

CONFIDENTIAL
ATARI
PRELIMINARY

2 of 12

MONITOR

Give to Gary

TITLE "MONITOR ***** MONITP.SRC ***** 3/9/79 ***** 4:00:00 P.M."

CONSTANT EQUATES

| | | | | |
|------|--------|---|------|------------------------------------|
| 0009 | PUTTXT | = | \$9 | "PUT TEXT RECORD" CIO COMMAND CODE |
| 0007 | GETCAR | = | \$7 | "GET CHARACTER" CIO COMMAND CODE |
| 0000 | PUTCAR | = | \$8 | "PUT CHARACTER" CIO COMMAND CODE |
| 0000 | INIMLL | = | \$00 | INITIAL MEM LO LOW BYTE |
| 0007 | INIMLH | = | \$07 | INITIAL MEM LO HIGH BYTE |
| 0098 | CR | = | \$98 | ASCII CARRIAGE RETURN |
| 0001 | GOOD | = | \$1 | GOOD STATUS CODE |
| 0057 | WRITE | = | \$57 | WRITE COMMAND |
| 0052 | READ | = | \$52 | READ COMMAND |
| 0053 | STATC | = | \$53 | STATUS COMMAND |
| 0000 | SEX | = | \$0 | SCREEN EDITOR IOCB INDEX |
| 007D | CLS | = | \$7D | CLEAR SCREEN CODE |
| 0092 | CTRLC | = | \$92 | KEYBOARD CODE FOR 'CONTROL C' |
| 0088 | EOF | = | \$88 | CASSETTE END OF FILE CODE |
| 0000 | LIR0 | = | \$0 | LONG I/O TYPE CODE |

| | | | | |
|------|-------|---|---------------------|----------------|
| 0004 | BUFFH | = | CASBUF+3/256 | |
| 0000 | BUFFL | = | -256+BUFFH+CASBUF+3 | BUFFER POINTER |

THE FOLLOWING EQUATES ARE IN THE CARTRIDGE ADDRESS SPACE.

B CARTRIDGE ADDR'S ARE 8000-9FFF (32K CONFIG. ONLY)

A CART. ADDR'S ARE A000-BFFF (32K CONFIG. ONLY)

A CART. ADDR'S ARE 8000-BFFF (48K CONFIG. ONLY)

| | | | | |
|------|--------|----|------|-----------------------------------|
| 00FA | CARTCB | == | 00FA | CARTRIDGE COLD START ADDRESS |
| 00FC | CART | == | 00FC | CARTRIDGE AVAILABLE FLAG BYTE |
| 00FD | CARTFO | == | 00FD | CARTRIDGE FLAG BYTE. BIT 0=FLAG1. |
| 00FE | CARTAD | == | 00FE | 2-BYTE CARTRIDGE START VECTOR |

CARTRIDGE FLAG ACTION DEFINITIONS

| BIT | ACTION IF SET |
|-----|---|
| 7 | SPECIAL -- DON'T POWER-UP, JUST RUN CARTRIDGE |
| 6-3 | NONE |
| 2 | RUN CARTRIDGE |
| 1 | NONE |
| 0 | BOOT DOS |

NOTE

1. IF BIT2 IS 0, GOTO BLACKBOARD MODE.

2. IF BIT0 SET, THE DISK WILL BE BOOTED BEFORE ANY
OTHER ACTION.

POWER-UP VECTOR

* = \$FFFC

PVECT WORD PWRUP POWER-UP VECTOR

ENTRY POINT VECTOR

* = BLASDV

I-471 4C 23 F2 JMP BIGNON BLACK BOARD VECTOR

* = WARMSTV

I-474 4C 06 90 JMP \$9006 WARM START VECTOR

* = COLDSTV

I-477 4C 00 90 JMP \$9000 COLD START VECTOR (\$9000 FOR RAM VECTOR WRITING)

* = \$9000

7000 20 0C 90 JSR \$900C
7003 4C 25 F1 JMP PWRUP (TO HANDLE RAM VECTOR WRITING)
7006 20 0C 90 JSR \$900C
9009 4C 1B F1 JMP RESET

* = MONOR0

HANDLER TABLE ENTRIES

| ADDRESS | LENGTH | BYTE | "P" |
|-------------|--------|------|---------|
| I-0E3 50 | | WORD | PRINTV |
| I-0E4 30 E4 | | WORD | PRINTV |
| I-0E4 43 | | BYTE | "C" |
| I-0E7 40 E4 | | WORD | CASETV |
| I-0E9 45 | | BYTE | "E" |
| I-0EA 00 E4 | | WORD | EDITRV |
| I-0EC 53 | | BYTE | "S" |
| I-0ED 10 E4 | | WORD | SCREENV |

I-0EF 4B BYTE "K"
I-0F0 20 E4 WORD KEYBDV

000E TBLLEN = IDENT-TBLENT-1 HANDLER TABLE LENGTH.

***** PRINT MESSAGES *****

I-0F2 7D 41 54 IDENT BYTE CLS, "ATARI COMPUTER - MEMO PAD", CR
I-0F5 41 52 49

I-0FB 20 43 4F
I-0FB 4D 80 85
I-0FE 54 45 52
I-101 20 2D 20
I-104 4D 45 4D
I-107 4F 20 50
I-10A 41 44 9B

00F0 IDENTH = IDENT/256
00F2 IDENTL = -256*IDENTH+IDENT SYSTEM I.D. MS0 POINTER

I-10D 42 4F 4F DERR BYTE "BOOT ERROR", CR

F110 54 20 45
F113 52 52 4F
F116 52 9B

00F1 DERRH = DERR/256
00D0 DERRL = -256*DERRH+DERR DISK ERROR MS0 POINTER

DEVICE/FILENAME SPECIFICATIONS

F11B 45 3A 9B OPNEDT = BYTE "OPEN SCREEN EDITOR" DEVICE SPEC

00F1 OPNH = OPNEDT/256
0018 OPNL = -256*OPNH+OPNEDT SCREEN EDITOR OPEN POINTER

***** RESET BUTTON ROUTINE STARTS HERE *****

F11B 7B RESET SET DISABLE IRQ INTERRUPTS
I-11C AD 44 03 LDA COLDST WERE WE IN MIDDLE OF COLDSTART?
I-11F D0 04 BNE PWRUP YES, GO TRY IT AGAIN
I-121 A9 FF LDA #FF
I-123 D0 03 BNE PWRUP1 SET WARM START FLAG

***** POWER UP ROUTINES START HERE *****

```

F125 78      PWRUP  BEI      DISABLE IRQ INTERRUPTS
F126 A9 00    LDA      #0      CLEAR WARMSTART FLAG
F128 B5 08    PWRUP1  STA      WARMST
F12A D8      CLD      CLEAR DECIMAL FLAG.
F12B A2 FF    LDX      #FF
F12D 9A      TXB
F12E 20 3F F2 JSR      SPECL  SET STACK POINTER
F131 20 81 F2 JSR      HARDI  CARTRIDGE SPECIAL CASE?
F134 A5 08    LDA      WARMST  DO HARDWARE INITIALIZATION
F136 D0 28    BNE      ZOBGRAM  IS IT WARMSTART?
                                YES, ONLY ZERO OS RAM

```

```

F138 A9 00    ZERORM  LDA      #0
F13A A0 08    LDY      #WARMST
F13C B5 04    STA      RAMLO
F13E B5 03    STA      RAMLO+1  INITIALIZE RAM POINTER
F140 91 04    CLRRAM  STA      (RAMLO),Y  CLEAR MEMORY LOC.
F142 C8      INY
F143 C0 00    CPY      #0      AT END OF PAGE?
F145 D0 F9    BNE      CLRRAM
F147 E6 03    INC      RAMLO+1  YES INCR PAGE POINTER
F149 A6 03    LDX      RAMLO+1
F14B E4 06    CPX      TRAMBZ
F14D D0 F1    BNE      CLRRAM  AT END OF MEM?
                                NO

```

```

F14F AD 72 E4 INITIALIZE DOBVEC TO POINT TO SIGNON (BLACKBOARD)
F152 B5 0A    LDA      BLKBDV+1
F154 AD 73 E4 STA      DOBVEC  USE BLACKBOARD VECTOR
F157 B5 08    LDA      BLKBDV+2  FOR DOBVEC
F159 A9 FF    STA      DOBVEC+1
F15B BD 44 02 LDA      #FF
F15E D0 13    STA      COLDBT  SET TO SHOW IN MIDDLE OF COLDBT
                                DO AROUND ZOBGRAM
F15E D0 13    BNE      ESTBCH

```

```

F160 A2 00    CLEAR OS RAM (FOR WARMSTART)
F162 BA      ZOBGRAM LDX      #0
F163 9D 00 02 ZOBRM2  TXA
F165 9D 00 03 STA      #200,X  CLEAR PAGES 2 AND 3
F167 CA      STA      #300,X
F16A D0 F7    DEX
F16C A2 10    BNE      ZOBRM2
F16E 93 00    LDX      #INITZBB
F170 E8      ZOBRM3  STA      0,X  CLEAR ZERO PAGE LOCATIONS INTZBB-7F
F171 10 FB    INX
F171 10 FB    BPL      ZOBRM3

```

```

F173 A9 02    ESTBCH  LDA      #LEDGE
F175 B5 52    STA      LMARGN
F177 A9 27    LDA      #REDGE
F179 B5 53    STA      RMARGN

```

```

F17B A2 25    MOVE VECTOR TABLE FROM ROM TO RAM
F17D BD 80 E4 OPBYB  LDX      #625
F180 9D 00 02 MOVVEC  LDA      VCTABL,X  ROM TABLE
F183 CA      STA      INTABB,X  TO RAM
F184 10 F7    DEX
F186 20 94 F2 BPL      MOVVEC
F189 5B      JSR      OSRAM  DO O.B. RAM SETUP
                                ENABLE IRQ INTERRUPTS
F189 5B      CLI

```

LINK HANDLERS

| | | | |
|---------------|--------|--------|------------------------|
| F18A A2 0E | LDX | #TBLEN | |
| F18C BD E3 F0 | NXTENT | LDA | TBLEN, X |
| F18F 9D 1A 03 | | STA | HATAB, X |
| F192 CA | DEX | | |
| F193 10 F7 | BPL | NXTENT | DONE WITH ALL ENTRIES? |

INTERROGATE CARTRIDGE ADDR. SPACE TO SEE WHICH CARTRIDGES THERE ARE

| | | | |
|---------------|--------|-------------|------------------------------|
| F195 A2 00 | LDX | #0 | |
| F197 B6 07 | BTX | TBTDAT | CLEAR "B" CART. FLAG |
| F199 B6 06 | BTX | TRAMZ | CLEAR "A" CART. FLAG |
| F19B AE E4 02 | LDX | RAMSIZ | |
| F19E E0 90 | CPX | #90 | RAM IN "B" CART. SLOT? |
| F1A0 B0 0A | BCS | ENDBCK | |
| F1A2 AD FC 9F | LDA | CART-\$2000 | NO. |
| F1A5 D0 05 | BNE | ENDBCK | CART. PLUGGED INTO "B" SLOT? |
| F1A7 E6 07 | INC | TBTDAT | YES, SET "B" CART. FLAG |
| F1A9 20 3C F2 | JBR | CBINI | INITIALIZE CARTRIDGE "B" |
| F1AC AE E4 02 | ENDBCK | LDX | RAMSIZ |
| F1AF E0 B0 | CPX | #B0 | RAM IN "A" CART. SLOT? |
| F1B1 B0 0A | BCS | ENDACK | |
| F1B3 AE FC BF | LDX | CART | NO. |
| F1B6 D0 05 | BNE | ENDACK | CART. PLUGGED INTO "A" SLOT? |
| F1B8 E6 06 | INC | TRAMZ | YES, SET "A" CART. FLAG |
| F1BA 20 39 F2 | JBR | CAINI | INITIALIZE CARTRIDGE "A" |

OPEN SCREEN EDITOR

| | | | |
|---------------|--------|----------|---|
| F1BD A9 03 | ENDACK | LDA | #3 |
| F1BF A2 00 | LDX | #SEX | |
| F1C1 9D 42 03 | STA | ICCOM, X | OPEN I/O COMMAND |
| F1C4 A9 18 | LDA | #OPNL | |
| F1C6 9D 44 03 | STA | ICBAL, X | |
| F1C9 A9 F1 | LDA | #OPNI | |
| F1CB 9D 45 03 | STA | ICBAH, X | SET BUFFER POINTER TO OPEN SCREEN EDITOR |
| F1CE A9 0C | LDA | #C | |
| F1D0 9D 4A 03 | STA | ICAX1, X | SET UP OPEN FOR INPUT/OUTPUT |
| F1D3 20 56 E4 | JBR | CIOV | GO TO CIO |
| F1D6 10 03 | BPL | SCRNOK | BR IF NO ERROR |
| F1DB 4C 25 F1 | JMP | PWRUP | RETRY PWRUP IF ERROR (SHOULD NEVER HAPPEN!) |
| F1DB E8 | SCRNOK | INX | SCREEN OK, SO WAIT FOR VBLANK TO |
| F1DC D0 FD | BNE | SCRNOK | BRING UP THE DISPLAY |
| F1DE C8 | INX | | |
| F1DF 10 FA | BPL | SCRNOK | |

DO CASSETTE BOOT

| | | | |
|---------------|-----|--------|-----------------------|
| F1E1 20 B2 F3 | JBR | CSBOOT | CHECK, BOOT, AND INIT |
|---------------|-----|--------|-----------------------|

```

CHECK TO SEE IF EITHER CARTRIDGE WANTS DISK BOOT
F1E4 A5 06 LDA TRAMBZ CHECK BOTH CARTRIDGES
F1E6 05 07 ORA TSTDAT
F1E8 F0 12 BEQ NOCART NEITHER CARTRIDGE LIVES
F1EA A5 06 LDA TRAMBZ "A" CART?
F1EC F0 03 BEQ NOA1 NO
F1EE AD FD BF LDA CARTF0 GET CARTRIDGE MODE FLAG
F1F1 A6 07 NOA1 LDX TSTDAT "B" CART?
F1F3 F0 03 BEQ NOB1 NO
F1F5 0D FD 9F ORA CARTF0-$2000 ADD OTHER FLAG
F1FB 29 01 NOB1 AND #1 DOES EITHER CART WANT BOOT?
F1FA F0 03 BEQ NOBOOT NO

```

```

DO DISK BOOT
F1FC 20 CF F2 NOCART JSR BOOT CHECK, BOOT, AND INIT

```

```

GO TO ONE OF THE CARTRIDGES IF THEY SO DESIRE
F1FF A9 00 NOBOOT LDA #0
F201 BD 44 03 STA COLDSTY RESET TO SHOW DONE WITH COLDSTART
F204 A5 06 LDA TRAMBZ "A" CART?
F206 F0 0A BEQ NOA2 NO
F208 AD FD BF LDA CARTF0 GET CARTRIDGE MODE FLAG
F20B 29 04 AND #4 DOES IT WANT TO RUN?
F20D F0 03 BEQ NOA2 NO
F20F 6C FA BF JMP (CARTCB) RUN "A" CARTRIDGE
F212 A5 07 NOA2 LDA TSTDAT "B" CART?
F214 F0 0A BEQ NOCAR2 NO
F216 AD FD 9F LDA CARTF0-$2000 GET "B" MODE FLAG
F219 29 04 AND #4 DOES IT WANT TO RUN?
F21B F0 DF BEQ NOCART NO
F21D 6C FA 9F JMP (CARTCB-$2000) RUN "B" CARTRIDGE

```

```

NO CARTRIDGES, OR NEITHER WANTS TO RUN,
SO GO TO DOBVEC (DOS, CASSETTE, OR BLACKBOARD)
F220 6C 0A 00 NOCAR2 JMP (DOBVEC)

```

```

PRINT SIGN-ON MESSAGE
F223 A2 F2 SIGNON LDX #IDENTL
F225 A0 F0 LDY #IDENTH
F227 20 B5 F3 JSR PUTLIN GO PUT SIGN-ON MSG ON SCREEN

```

```

BLACKBOARD ROUTINE
F22A 20 30 F2 BLACKB JSR BLKB2 "JSR EQETCH"
F22D 4C 2A F2 JMP BLACKB FOREVER
F230 AD 05 E4 BLKB2 LDA EDITRV+5 HIGH BYTE
F233 48 PHA
F234 AD 04 E4 LDA EDITRV+4 LOW BYTE
F237 48 PHA
F238 60 RTS SIMULATES "JMP (EDITRV)"

```

```

CARTRIDGE INITIALIZATION INDIRECT JUMPS
F239 6C FE BF CAINI JMP (CARTAD)
F23C 6C FE 9F CBINI JMP (CARTAD-$2000)

```


SUBROUTINES

CHECK FOR HOW MUCH RAM & SPECIAL CARTRIDGE CASE.
IF SPECIAL CARTRIDGE CASE, DON'T GO BACK -- GO TO CART.

| Address | Op | Op2 | Op3 | Op4 | Op5 | Op6 | Op7 | Op8 | Op9 | Op10 | Op11 | Op12 | Op13 | Op14 | Op15 | Op16 | Op17 | Op18 | Op19 | Op20 | Op21 | Op22 | Op23 | Op24 | Op25 | Op26 | Op27 | Op28 | Op29 | Op30 | Op31 | Op32 | Op33 | Op34 | Op35 | Op36 | Op37 | Op38 | Op39 | Op40 | Op41 | Op42 | Op43 | Op44 | Op45 | Op46 | Op47 | Op48 | Op49 | Op50 | Op51 | Op52 | Op53 | Op54 | Op55 | Op56 | Op57 | Op58 | Op59 | Op60 | Op61 | Op62 | Op63 | Op64 | Op65 | Op66 | Op67 | Op68 | Op69 | Op70 | Op71 | Op72 | Op73 | Op74 | Op75 | Op76 | Op77 | Op78 | Op79 | Op80 | Op81 | Op82 | Op83 | Op84 | Op85 | Op86 | Op87 | Op88 | Op89 | Op90 | Op91 | Op92 | Op93 | Op94 | Op95 | Op96 | Op97 | Op98 | Op99 | Op100 | Op101 | Op102 | Op103 | Op104 | Op105 | Op106 | Op107 | Op108 | Op109 | Op110 | Op111 | Op112 | Op113 | Op114 | Op115 | Op116 | Op117 | Op118 | Op119 | Op120 | Op121 | Op122 | Op123 | Op124 | Op125 | Op126 | Op127 | Op128 | Op129 | Op130 | Op131 | Op132 | Op133 | Op134 | Op135 | Op136 | Op137 | Op138 | Op139 | Op140 | Op141 | Op142 | Op143 | Op144 | Op145 | Op146 | Op147 | Op148 | Op149 | Op150 | Op151 | Op152 | Op153 | Op154 | Op155 | Op156 | Op157 | Op158 | Op159 | Op160 | Op161 | Op162 | Op163 | Op164 | Op165 | Op166 | Op167 | Op168 | Op169 | Op170 | Op171 | Op172 | Op173 | Op174 | Op175 | Op176 | Op177 | Op178 | Op179 | Op180 | Op181 | Op182 | Op183 | Op184 | Op185 | Op186 | Op187 | Op188 | Op189 | Op190 | Op191 | Op192 | Op193 | Op194 | Op195 | Op196 | Op197 | Op198 | Op199 | Op200 | Op201 | Op202 | Op203 | Op204 | Op205 | Op206 | Op207 | Op208 | Op209 | Op210 | Op211 | Op212 | Op213 | Op214 | Op215 | Op216 | Op217 | Op218 | Op219 | Op220 | Op221 | Op222 | Op223 | Op224 | Op225 | Op226 | Op227 | Op228 | Op229 | Op230 | Op231 | Op232 | Op233 | Op234 | Op235 | Op236 | Op237 | Op238 | Op239 | Op240 | Op241 | Op242 | Op243 | Op244 | Op245 | Op246 | Op247 | Op248 | Op249 | Op250 | Op251 | Op252 | Op253 | Op254 | Op255 | Op256 | Op257 | Op258 | Op259 | Op260 | Op261 | Op262 | Op263 | Op264 | Op265 | Op266 | Op267 | Op268 | Op269 | Op270 | Op271 | Op272 | Op273 | Op274 | Op275 | Op276 | Op277 | Op278 | Op279 | Op280 | Op281 | Op282 | Op283 | Op284 | Op285 | Op286 | Op287 | Op288 | Op289 | Op290 | Op291 | Op292 | Op293 | Op294 | Op295 | Op296 | Op297 | Op298 | Op299 | Op300 | Op301 | Op302 | Op303 | Op304 | Op305 | Op306 | Op307 | Op308 | Op309 | Op310 | Op311 | Op312 | Op313 | Op314 | Op315 | Op316 | Op317 | Op318 | Op319 | Op320 | Op321 | Op322 | Op323 | Op324 | Op325 | Op326 | Op327 | Op328 | Op329 | Op330 | Op331 | Op332 | Op333 | Op334 | Op335 | Op336 | Op337 | Op338 | Op339 | Op340 | Op341 | Op342 | Op343 | Op344 | Op345 | Op346 | Op347 | Op348 | Op349 | Op350 | Op351 | Op352 | Op353 | Op354 | Op355 | Op356 | Op357 | Op358 | Op359 | Op360 | Op361 | Op362 | Op363 | Op364 | Op365 | Op366 | Op367 | Op368 | Op369 | Op370 | Op371 | Op372 | Op373 | Op374 | Op375 | Op376 | Op377 | Op378 | Op379 | Op380 | Op381 | Op382 | Op383 | Op384 | Op385 | Op386 | Op387 | Op388 | Op389 | Op390 | Op391 | Op392 | Op393 | Op394 | Op395 | Op396 | Op397 | Op398 | Op399 | Op400 | Op401 | Op402 | Op403 | Op404 | Op405 | Op406 | Op407 | Op408 | Op409 | Op410 | Op411 | Op412 | Op413 | Op414 | Op415 | Op416 | Op417 | Op418 | Op419 | Op420 | Op421 | Op422 | Op423 | Op424 | Op425 | Op426 | Op427 | Op428 | Op429 | Op430 | Op431 | Op432 | Op433 | Op434 | Op435 | Op436 | Op437 | Op438 | Op439 | Op440 | Op441 | Op442 | Op443 | Op444 | Op445 | Op446 | Op447 | Op448 | Op449 | Op450 | Op451 | Op452 | Op453 | Op454 | Op455 | Op456 | Op457 | Op458 | Op459 | Op460 | Op461 | Op462 | Op463 | Op464 | Op465 | Op466 | Op467 | Op468 | Op469 | Op470 | Op471 | Op472 | Op473 | Op474 | Op475 | Op476 | Op477 | Op478 | Op479 | Op480 | Op481 | Op482 | Op483 | Op484 | Op485 | Op486 | Op487 | Op488 | Op489 | Op490 | Op491 | Op492 | Op493 | Op494 | Op495 | Op496 | Op497 | Op498 | Op499 | Op500 | Op501 | Op502 | Op503 | Op504 | Op505 | Op506 | Op507 | Op508 | Op509 | Op510 | Op511 | Op512 | Op513 | Op514 | Op515 | Op516 | Op517 | Op518 | Op519 | Op520 | Op521 | Op522 | Op523 | Op524 | Op525 | Op526 | Op527 | Op528 | Op529 | Op530 | Op531 | Op532 | Op533 | Op534 | Op535 | Op536 | Op537 | Op538 | Op539 | Op540 | Op541 | Op542 | Op543 | Op544 | Op545 | Op546 | Op547 | Op548 | Op549 | Op550 | Op551 | Op552 | Op553 | Op554 | Op555 | Op556 | Op557 | Op558 | Op559 | Op560 | Op561 | Op562 | Op563 | Op564 | Op565 | Op566 | Op567 | Op568 | Op569 | Op570 | Op571 | Op572 | Op573 | Op574 | Op575 | Op576 | Op577 | Op578 | Op579 | Op580 | Op581 | Op582 | Op583 | Op584 | Op585 | Op586 | Op587 | Op588 | Op589 | Op590 | Op591 | Op592 | Op593 | Op594 | Op595 | Op596 | Op597 | Op598 | Op599 | Op600 | Op601 | Op602 | Op603 | Op604 | Op605 | Op606 | Op607 | Op608 | Op609 | Op610 | Op611 | Op612 | Op613 | Op614 | Op615 | Op616 | Op617 | Op618 | Op619 | Op620 | Op621 | Op622 | Op623 | Op624 | Op625 | Op626 | Op627 | Op628 | Op629 | Op630 | Op631 | Op632 | Op633 | Op634 | Op635 | Op636 | Op637 | Op638 | Op639 | Op640 | Op641 | Op642 | Op643 | Op644 | Op645 | Op646 | Op647 | Op648 | Op649 | Op650 | Op651 | Op652 | Op653 | Op654 | Op655 | Op656 | Op657 | Op658 | Op659 | Op660 | Op661 | Op662 | Op663 | Op664 | Op665 | Op666 | Op667 | Op668 | Op669 | Op670 | Op671 | Op672 | Op673 | Op674 | Op675 | Op676 | Op677 | Op678 | Op679 | Op680 | Op681 | Op682 | Op683 | Op684 | Op685 | Op686 | Op687 | Op688 | Op689 | Op690 | Op691 | Op692 | Op693 | Op694 | Op695 | Op696 | Op697 | Op698 | Op699 | Op700 | Op701 | Op702 | Op703 | Op704 | Op705 | Op706 | Op707 | Op708 | Op709 | Op710 | Op711 | Op712 | Op713 | Op714 | Op715 | Op716 | Op717 | Op718 | Op719 | Op720 | Op721 | Op722 | Op723 | Op724 | Op725 | Op726 | Op727 | Op728 | Op729 | Op730 | Op731 | Op732 | Op733 | Op734 | Op735 | Op736 | Op737 | Op738 | Op739 | Op740 | Op741 | Op742 | Op743 | Op744 | Op745 | Op746 | Op747 | Op748 | Op749 | Op750 | Op751 | Op752 | Op753 | Op754 | Op755 | Op756 | Op757 | Op758 | Op759 | Op760 | Op761 | Op762 | Op763 | Op764 | Op765 | Op766 | Op767 | Op768 | Op769 | Op770 | Op771 | Op772 | Op773 | Op774 | Op775 | Op776 | Op777 | Op778 | Op779 | Op780 | Op781 | Op782 | Op783 | Op784 | Op785 | Op786 | Op787 | Op788 | Op789 | Op790 | Op791 | Op792 | Op793 | Op794 | Op795 | Op796 | Op797 | Op798 | Op799 | Op800 | Op801 | Op802 | Op803 | Op804 | Op805 | Op806 | Op807 | Op808 | Op809 | Op810 | Op811 | Op812 | Op813 | Op814 | Op815 | Op816 | Op817 | Op818 | Op819 | Op820 | Op821 | Op822 | Op823 | Op824 | Op825 | Op826 | Op827 | Op828 | Op829 | Op830 | Op831 | Op832 | Op833 | Op834 | Op835 | Op836 | Op837 | Op838 | Op839 | Op840 | Op841 | Op842 | Op843 | Op844 | Op845 | Op846 | Op847 | Op848 | Op849 | Op850 | Op851 | Op852 | Op853 | Op854 | Op855 | Op856 | Op857 | Op858 | Op859 | Op860 | Op861 | Op862 | Op863 | Op864 | Op865 | Op866 | Op867 | Op868 | Op869 | Op870 | Op871 | Op872 | Op873 | Op874 | Op875 | Op876 | Op877 | Op878 | Op879 | Op880 | Op881 | Op882 | Op883 | Op884 | Op885 | Op886 | Op887 | Op888 | Op889 | Op890 | Op891 | Op892 | Op893 | Op894 | Op895 | Op896 | Op897 | Op898 | Op899 | Op900 | Op901 | Op902 | Op903 | Op904 | Op905 | Op906 | Op907 | Op908 | Op909 | Op910 | Op911 | Op912 | Op913 | Op914 | Op915 | Op916 | Op917 | Op918 | Op919 | Op920 | Op921 | Op922 | Op923 | Op924 | Op925 | Op926 | Op927 | Op928 | Op929 | Op930 | Op931 | Op932 | Op933 | Op934 | Op935 | Op936 | Op937 | Op938 | Op939 | Op940 | Op941 | Op942 | Op943 | Op944 | Op945 | Op946 | Op947 | Op948 | Op949 | Op950 | Op951 | Op952 | Op953 | Op954 | Op955 | Op956 | Op957 | Op958 | Op959 | Op960 | Op961 | Op962 | Op963 | Op964 | Op965 | Op966 | Op967 | Op968 | Op969 | Op970 | Op971 | Op972 | Op973 | Op974 | Op975 | Op976 | Op977 | Op978 | Op979 | Op980 | Op981 | Op982 | Op983 | Op984 | Op985 | Op986 | Op987 | Op988 | Op989 | Op990 | Op991 | Op992 | Op993 | Op994 | Op995 | Op996 | Op997 | Op998 | Op999 |
|---------|----|-----|-----|-------|----------|------|----------------------------|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| I-23F | AD | FC | BF | BPECL | LDA | CART | CHECK FOR RAM OR CART | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| F242 | D0 | 12 | | BNE | ENSPEC | | GO IF NOTHING OR MAYBE RAM | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| F244 | EE | FC | BF | INC | CART | | NOW DO RAM CHECK | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| F247 | AD | FC | BF | LDA | CART | | IS IT RAM? | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| I-24A | D0 | 0A | | BNE | ENSPEC | | NO | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| I-24C | AD | FD | BF | LDA | CARTFO | | YES, | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| I-24F | 29 | 80 | | AND | ##80 | | MASK OFF SPECIAL BIT | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| F251 | F0 | 03 | | BEQ | ENSPEC | | BIT BET? | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| F253 | 4C | FE | BF | JMP | (CARTAD) | | YES, GO RUN CARTRIDGE | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

CHECK FOR AMOUNT OF RAM

| Address | Op | Op2 | Op3 | Op4 | Op5 | Op6 | Op7 | Op8 | Op9 | Op10 | Op11 | Op12 | Op13 | Op14 | Op15 | Op16 | Op17 | Op18 | Op19 | Op20 | Op21 | Op22 | Op23 | Op24 | Op25 | Op26 | Op27 | Op28 | Op29 | Op30 | Op31 | Op32 | Op33 | Op34 | Op35 | Op36 | Op37 | Op38 | Op39 | Op40 | Op41 | Op42 | Op43 | Op44 | Op45 | Op46 | Op47 | Op48 | Op49 | Op50 | Op51 | Op52 | Op53 | Op54 | Op55 | Op56 | Op57 | Op58 | Op59 | Op60 | Op61 | Op62 | Op63 | Op64 | Op65 | Op66 | Op67 | Op68 | Op69 | Op70 | Op71 | Op72 | Op73 | Op74 | Op75 | Op76 | Op77 | Op78 | Op79 | Op80 | Op81 | Op82 | Op83 | Op84 | Op85 | Op86 | Op87 | Op88 | Op89 | Op90 | Op91 | Op92 | Op93 | Op94 | Op95 | Op96 | Op97 | Op98 | Op99 | Op100 | Op101 | Op102 | Op103 | Op104 | Op105 | Op106 | Op107 | Op108 | Op109 | Op110 | Op111 | Op112 | Op113 | Op114 | Op115 | Op116 | Op117 | Op118 | Op119 | Op120 | Op121 | Op122 | Op123 | Op124 | Op125 | Op126 | Op127 | Op128 | Op129 | Op130 | Op131 | Op132 | Op133 | Op134 | Op135 | Op136 | Op137 | Op138 | Op139 | Op140 | Op141 | Op142 | Op143 | Op144 | Op145 | Op146 | Op147 | Op148 | Op149 | Op150 | Op151 | Op152 | Op153 | Op154 | Op155 | Op156 | Op157 | Op158 | Op159 | Op160 | Op161 | Op162 | Op163 | Op164 | Op165 | Op166 | Op167 | Op16 |
|---------|----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|------|
|---------|----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|------|

| | | | |
|------------|--------|--------|--------------------------|
| 127C 85 06 | STA | TRAMBZ | INCR. RAM POINTER BY 4K. |
| 127E D0 E3 | BNE | HOWMCH | GO FIND HOW MUCH RAM. |
| 1280 60 | ENDRAM | RTS | |

HARDWARE INITIALIZATION

| | | | |
|---------------|--------|--------|-----------|
| 1281 A9 00 | HARDI | LDA | #0 |
| 1283 AA | TAX | | |
| 1284 9D 00 D0 | CLRCHP | STA | \$D000, X |
| 1287 9D 00 D4 | | STA | \$D400, X |
| 128A 9D 00 D2 | | STA | \$D200, X |
| 128D 9D 00 D3 | | STA | \$D300, X |
| 1290 E8 | INX | | |
| 1291 D0 F1 | BNE | CLRCHP | |
| 1293 60 | RTS | | |

0.5. RAM SETUP

| | | | | |
|---------------|-------|-----|------------|------------------------------|
| 1294 C6 11 | DSRAM | DEC | BRKKEY | TURN OFF BREAK KEY FLAG |
| 1296 A5 06 | | LDA | TRAMBZ | READ RAM SIZE IN TEMP REG |
| 1298 8D E4 02 | | STA | RAMBIZ | SAVE IT IN RAM SIZE |
| 129B 8D E6 02 | | STA | MENTOP+1 | INIT. MENTOP ADDR HI BYTE |
| 129E A9 00 | | LDA | #0 | |
| 12A0 8D E5 02 | | STA | MENTOP | INIT. MENTOP ADDR LO BYTE |
| 12A3 A9 00 | | LDA | #INIML | |
| 12A5 8D E7 02 | | STA | MEMLO | |
| 12AB A9 07 | | LDA | #INIMH | |
| 12AA 8D E8 02 | | STA | MEMLO+1 | INITIALIZE MEMLO ADDR VECTOR |
| 12AD 20 0C E4 | | JSR | EDITRV+8C | EDITOR INIT. |
| 12B0 20 1C E4 | | JSR | SCREENV+8C | SCREEN INIT. |
| 12B3 20 2C E4 | | JSR | KEYBDV+8C | KEYBOARD INIT. |
| 12B6 20 3C E4 | | JSR | PRINTV+8C | PRINTER HANDLER INIT |
| 12B9 20 4C E4 | | JSR | CASSETV+8C | CASSETTE HANDLER INIT |
| 12BC 20 6E E4 | | JSR | CIOINV | CIO INIT. |
| 12BF 20 65 E4 | | JSR | SIOINV | SIO INIT. |
| 12C2 20 6B E4 | | JSR | INTINV | INTERRUPT HANDLER INIT. |
| 12C5 AD 1F D0 | | LDA | CONBOL | |
| 12C8 29 01 | | AND | #01 | |
| 12CA D0 02 | | BNE | NOKEY | GAME START KEY DEPRESSED? |
| 12CC E6 4A | | INC | CKEY | YES, SET KEY FLAG. |
| 12CE 60 | NOKEY | RTS | | |

DO BOOT OF DISK

| | | | | |
|---------------|--------|-----|--------|----------------------------|
| 12CF A5 08 | BOOT | LDA | WARMST | |
| 12D1 F0 0A | | BEG | NOWARM | WARM START? |
| 12D3 A5 09 | | LDA | BOOT? | YES. |
| 12D5 29 01 | | AND | #1 | |
| 12D7 F0 03 | | BEG | NOINIT | VALID BOOT? |
| 12D9 20 7E F3 | | JSR | DINI | YES, RE-INIT. DOS SOFTWARE |
| 12DC 60 | NOINIT | RTS | | |
| 12DD A9 01 | NOWARM | LDA | #1 | |
| 12DF 8D 01 03 | | STA | DUNIT | ASSIGN DISK DRIVE NO. |
| 12E2 A9 53 | | LDA | #STATC | |

| | | | |
|---------------|--------|-------------|--|
| 12E4 8D 02 03 | STA | DCOMND | SET UP STATUS COMMAND |
| 12E7 20 53 E4 | JSR | DSKINV | GO DO DISK STATUS |
| 12EA 10 01 | BPL | DOBOOT | IS STATUS FROM SIO GOOD? |
| 12EC 60 | RTS | | NO. GO BACK WITH BAD BOOT STATUS |
| 12ED A9 00 | DOBOOT | LDA | #0 |
| 12EF 8D 0B 03 | STA | DAUX2 | |
| 12F2 A9 01 | LDA | #1 | |
| 12F4 8D 0A 03 | STA | DAUX1 | SET SECTOR # TO 1. |
| 12F7 A9 00 | LDA | #BUFFL | |
| 12F9 8D 04 03 | STA | DBUFFLQ | |
| 12FC A9 04 | LDA | #BUFFH | |
| 12FE 8D 05 03 | STA | DBUFFH | SET UP BUFFER ADDR |
| 1301 20 9D F3 | SECT1 | JSR | GETSEC |
| 1304 10 0B | BPL | ALLSEC | GET SECTOR |
| 1306 20 81 F3 | BADDSK | JSR | STATUS O.K. ? |
| 1309 A5 4B | LDA | DSKRDE | NO. GO PRINT DISK READ ERROR |
| 130B F0 E0 | BEQ | CASSET | |
| 130D 60 | RTS | DOBOOT | CASSETTE BOOT? |
| 130E A2 03 | ALLSEC | LDX | YES, QUIT |
| 1310 8D 00 04 | RDBYTE | LDA | #3 |
| 1313 9D 40 02 | STA | CASBUF+3, X | READ A BUFFER BYTE |
| 1316 CA | DEX | DFLA0B, X | STORE IT |
| 1317 10 F7 | BPL | RDBYTE | |
| 1319 AD 42 02 | LDA | BOOTAD | DONE WITH 4 BYTE TRANSFER ? |
| 131C 85 04 | STA | RAMLO | YES. |
| 131E AD 43 02 | LDA | BOOTAD+1 | |
| 1321 85 05 | STA | RAMLO+1 | PUT BOOT ADDR INTO 2. PAGE RAM |
| 1323 AD 04 04 | LDA | CASBUF+7 | |
| 1326 85 00 | STA | DOSINI | ESTABLISH DOS INIT ADDRESS |
| 1328 AD 05 04 | LDA | CASBUF+8 | |
| 132B 85 0D | STA | DOSINI+1 | |
| 132D A0 7F | MVBUFF | LDY | |
| 132F 89 00 04 | MVNXB | LDA | #7F |
| 1332 91 04 | STA | CASBUF+3, Y | YES, SET BYTE COUNT |
| 1334 88 | DEY | (RAMLO), Y | MOVE A BYTE FROM SECTOR BUFFER TO BOOT ADDR. |
| 1335 10 F8 | BPL | MVNXB | |
| 1337 18 | CLC | | DONE ? |
| 1338 A5 04 | LDA | RAMLO | YES. |
| 133A 69 80 | ADC | #80 | |
| 133C 85 04 | STA | RAMLO | |
| 133E A5 05 | LDA | RAMLO+1 | |
| 1340 A9 00 | ADC | #0 | |
| 1342 85 05 | STA | RAMLO+1 | INCR BOOT LOADER BUFFER POINTER. |
| 1344 CE 41 02 | DEC | DBSECT | |
| 1347 F0 11 | BEQ | ENBOOT | DECR # OF SECTORS. |
| 1349 EE 0A 03 | INC | DAUX1 | MORE SECTORS ? |
| 134C 20 9D F3 | SECTX | JSR | YES, INCR SECTOR # |
| 134F 10 DC | BPL | GETSEC | GO GET SECTOR |
| 1351 20 81 F3 | JSR | MVBUFF | STATUS O.K. ? |
| 1354 A5 4B | LDA | DSKRDE | NO. GO PRINT DISK READ ERROR |
| 1356 D0 AE | BNE | CASSET | |
| 1358 F0 F2 | BEQ | BADDSK | IF CASSETTE, QUIT. |
| 135A A5 4B | ENBOOT | SECTX | IF DISK, TRY SECTOR AGAIN. |
| 135C F0 03 | LDA | CASSET | |
| 135E 20 9D F3 | BEQ | XBOOT | CASSETTE BOOT ? |
| 1361 20 6C F3 | JSR | GETSEC | YES, GET EOF RECORD, BUT DON'T USE IT. |
| 1364 80 A0 | BCS | BLOAD | GO EXECUTE BOOT LOADER |
| 1366 20 7E F3 | JSR | BADDSK | IF BAD BOOT, DO IT OVER AGAIN |
| 1369 E6 09 | INC | DINI | GO INIT. SOFTWARE |
| | | BOOT? | SHOW BOOT SUCCESS |

```

F36B 60          RTS
F36C 1B          BLDAD CLC
F36D AD 42 02    LDA     BOOTAD
F370 69 06        ADC     #6
F372 85 04        STA     RAMLO
F374 AD 43 02    LDA     BOOTAD+1
F377 69 00        ADC     #0
F379 85 05        STA     RAMLO+1    PUT START ADDR OF BOOTLOADER INTO RAM
F37B 6C 04 00    JMP     (RAMLO)
F37E 6C 0C 00    DINI    JMP     (DOSINI)

```

DISPLAY DISK READ ERROR MSG

```

F381 A2 0D        DSKRDE LDX     #DERRL
F383 A0 F1        LDY     #DERRH

```

PUT LINE ON SCREEN AT PRESENT CURSOR POSITION

X-REG -- LO BYTE, BEGIN ADDR OF LINE
Y-REG -- HI BYTE, BEGIN ADDR OF LINE

```

F385 8A          PUTLIN TXA
F386 A2 00        LDX     #SEX
F388 9D 44 03     STA     ICBAL,X
F38B 98          TYA
F38C 9D 45 03     STA     ICBAL,X    GET UP ADDR OF BEGIN OF LINE
F38F A9 09        LDA     #PUTTXT
F391 9D 42 03     STA     ICCOM,X    "PUT TEXT RECORD" COMMAND
F394 A9 FF        LDA     #6FF
F396 9D 48 03     STA     ICBLL,X    SET BUFFER LENGTH
F399 20 56 E4     JSR     CIOV    PUT LINE ON SCREEN
F39C 60          RTS

```

GET SECTOR FROM DISK 0

```

F39D A5 4B        GETSEC LDA     CASSET
F39F F0 03        BEQ     DISKM    CASSETTE BOOT ?
F3A1 4C 7A E4     JMP     RBLOKV    YES, GO TO READ BLOCK ROUTINE
F3A4 A9 52        DISKM  LDA     #READ
F3A6 8D 02 03     STA     DCOMND    GET READ SECTOR COMMAND
F3A9 A9 01        LDA     #1
F3AB 8D 01 03     STA     DUNIT    GET DRIVE NO. TO DRIVE 0
F3AE 20 53 E4     JSR     DSKINV    GET SECTOR
F3B1 60          RTS

```

DO CHECK FOR CASSETTE BOOT & IF SO, DO BOOT

```

F3B2 A5 08        CSBOOT LDA     WARMST    WARMSTART?
F3B4 F0 0A        BEQ     CSBOT2    NO
F3B6 A5 09        LDA     BOOT?    GET BOOT FLAG

```

| | | | |
|----------------|--------|----------|------------------------------------|
| I-308 29 02 | AND | #2 | WAS CASSETTE BOOT SUCCESFULL? |
| I-30A F0 03 | BEG | NOC8B2 | NO |
| I-30C 20 E1 F3 | JSR | CINI | YES, INIT CASSETTE SOFTWARE |
| I-30F 60 | NOC8B2 | RTB | |
| I-3C0 A5 4A | C8BOT2 | LDA | CKEY |
| I-3C2 F0 1C | BEG | NOC8BT | "C" KEY FLAG SET ? |
| I-3C4 A9 80 | LDA | #80 | YES, |
| I-3C6 85 3E | STA | FTYPE | SET LONG IRO TYPE |
| I-3C8 E6 4B | INC | CAS8BT | SET CASSETTE BOOT FLAG |
| I-3CA 20 7D E4 | JSR | C8OPIV | OPEN CASSETTE FOR INPUT |
| I-3CD 20 01 F3 | JSR | SECT1 | DO BOOT & INIT |
| I-3D0 A9 00 | LDA | #0 | |
| I-3D2 85 4B | STA | CAS8BT | REBET CASSETTE BOOT FLAG |
| I-3D4 85 4A | STA | CKEY | CLEAR KEY FLAG |
| I-3D6 06 09 | ASL | BOOT? | SHIFT BOOT FLAG (NOW=2 IF SUCCESS) |
| I-3D8 A5 0C | LDA | DOSINI | |
| I-3DA 85 02 | STA | CASINI | MOVE INIT ADDRESS FOR CASSETTE |
| I-3DC A5 0D | LDA | DOSINI+1 | |
| I-3DE 85 03 | STA | CASINI+1 | |
| I-3E0 60 | NOC8BT | RTB | |

I-3E1 6C 02 00 CINI JMP (CASINI) INIT CASSETTE

SPARE BYTE OR MODULE TOO LONG FLAG

I-3E4 CRNTPC = *

0014 00 MONBPR * - 014 ADDORG-CRNTPC MONITP TOO LONG

END

SYMBOL TABLE

| | | | | | | | |
|---------|------|---------|------|---------|------|--------|------|
| ADDRESS | 030E | ADDRESS | 0064 | APP | D800 | ALLPRT | D208 |
| ATTACH | F30E | ANTIC | D400 | APPEND | 0001 | APPMHI | 000E |
| ATTACH | 02FB | ATAN | BE43 | ATTRACT | 004D | AUDC1 | D201 |
| AUDC1 | D203 | AUDC3 | D205 | AUDC4 | D207 | AUDCTL | D208 |
| AUDF1 | D200 | AUDF2 | D202 | AUDF3 | D204 | AUDF4 | D206 |
| BADCHK | F306 | BADIOC | 0086 | BADMOD | 0091 | BFENHI | 0035 |
| BEENLD | 0034 | BITMAK | 006E | BLACKB | F22A | BLIM | 028A |
| BLKBD | F230 | BLKBDV | E471 | BLOAD | F36C | BOOT | F2CF |
| BOOT? | 0009 | BOOTAD | 0242 | BOTSCR | 02BF | BPTR | 003D |
| BRKART | 0080 | BRKKEY | 0011 | BUADDR | 0015 | BUFCNT | 0068 |
| BUFL | 0004 | BUFL | 0000 | BUFRFL | 003B | BUFRHI | 0033 |
| BUFRLO | 0032 | BUFRTR | 006C | CAINI | F239 | CART | BFFC |
| CARTAD | BFFE | CARTCS | BFFA | CARTFQ | BFFD | CASQUE | 03FD |
| CASCTV | E440 | CASFL0 | 030F | CASINI | 0002 | CASORQ | EF41 |
| CASRTI | 004B | CASSET | 0043 | CAUX1 | 023C | CAUX2 | 023D |
| CBADHI | 02EF | CBAVDL | 02EE | CBINI | F23C | CCOMND | 023B |
| CDEVIC | 023A | CDTMA1 | 0226 | CDTMA2 | 0228 | CDTHF3 | 022A |
| CDTHF4 | 022C | CDTHF5 | 022E | CDTHV1 | 0218 | CDTHV2 | 021A |
| CDTHV3 | 021C | CDTHV4 | 021E | CDTHV5 | 0220 | CH | 02FC |
| CH | 02F2 | CHACT | 02F3 | CHACTL | D401 | CHAR | 02FA |
| CHIAM | 02FA | CHBASE | D409 | CHKERR | 00BF | CHKENT | 003B |
| CHKRSH | 0031 | CHORO | E000 | CINI | F3E1 | CIOCHR | 002F |
| CTDINV | E46E | CIOORG | E4A6 | CIOV | E4B6 | CIX | 00F2 |
| CKY | 004A | CLOBE | 000C | CLRCHP | F2B4 | CLRRAM | F140 |
| CLSY | 007D | COLAC | 0072 | COLBK | D01A | COLCRS | 0055 |
| COLDET | 0244 | COLD8V | E477 | COLINC | 007A | COLOR0 | 02C4 |
| COLOR1 | 02C5 | COLOR2 | 02C6 | COLOR3 | 02C7 | COLOR4 | 02C8 |
| COLPFO | D016 | COLPF1 | D017 | COLPF2 | D018 | COLPF3 | D019 |
| COLPH0 | D012 | COLPH1 | D013 | COLPH2 | D014 | COLPH3 | D015 |
| COLRSH | 004F | CONBOL | D01F | COS | B073 | COUNTR | 007E |
| CR | 009B | CRETRY | 0036 | CRITIC | 0042 | CRNTPC | F3E4 |
| CRSHNI | 02F0 | CRSROR | 008D | CSBOOT | F3B2 | CSBOT2 | F3C0 |
| CRSHIV | E47D | CSTAT | 0288 | CTIA | D000 | CTRLC | 0092 |
| DAUX1 | 030A | DAUX2 | 030B | DBSECT | 0241 | DBUFHI | 0305 |
| DBUFLO | 0304 | DBYTH1 | 0309 | DBYTLO | 0308 | DCB | 0300 |
| DCOMND | 0302 | DDEVIC | 0300 | DEQFLO | 00FB | DEQON | 0006 |
| DELETE | 0021 | DELTA0 | 0077 | DELTA1 | 0076 | DERR | F10D |
| DERR1 | 00F1 | DERRL | 000D | DERROR | 0090 | DFLA08 | 0240 |
| DERT | 00F1 | DINDEX | 0057 | DINI | F37E | DIRECT | 0002 |
| DISK | 0044 | DISKIV | E450 | DISKM | F3A4 | DISPLY | 0053 |
| DISINH | D403 | DLTBL | D402 | DMACTL | D400 | DMASK | 02A0 |
| DMACK | 008B | DOBOOT | F2E0 | DOBINI | 000C | DOBVEC | 000A |
| DIAMN | 0011 | DRETRY | 0037 | DRKMSK | 004E | DSKFMS | 0018 |
| DSKINV | E453 | DSKOR0 | EDEA | DSKRDE | F3B1 | DSKTIM | 0246 |
| DSKUTL | 001A | DSPFLO | 02FE | DSTAT | 004C | DSTATB | 0303 |
| DTINH0 | 0306 | DUNIT | 0301 | DUNUSE | 0307 | DVSTAT | 02EA |
| EDITRV | E400 | EEXP | 00ED | ENBOOT | F35A | ENDACK | F1BD |
| ENDHCK | F1AC | ENDPT | 0074 | ENDRAM | F2B0 | ENSPEC | F256 |
| EOF | 008B | EOFERR | 008B | ERRFLO | 023F | ESCFLO | 02A2 |
| EXPON | 00EF | ESTSCH | F173 | EXP | DDC0 | EXP10 | DDCC |
| FADD | DA66 | FASC | D8E6 | FCHRFL | 00F0 | FDIV | DB28 |
| FILL | 003F | FILDAT | 02FD | FILFLO | 02B7 | FILLIN | 0012 |
| FLKOP | DDBD | FLDOR | DD89 | FLDIP | DD9C | FLDIR | DD98 |
| FLPTH | 00FC | FMOVE | DD86 | FMSZP0 | 0043 | FMUL | DAD8 |
| FNCMIT | 0072 | FDRMAT | 0022 | FPI | D9D2 | FPREC | 0006 |
| FPSCR | 05E6 | FPSCR1 | 05EC | FPTR2 | 00FE | FRO | 00D4 |
| FR | 00E0 | FR2 | 00E6 | FRE | 00DA | FREQ | 0040 |
| FRMRR | 008C | FRX | 00EC | FSCR | 05E6 | FSCR1 | 05EC |
| FRTOP | DDAB | FSTOR | DDA7 | FSUB | DA60 | FTYPE | 003E |

| | | | | | | | |
|---------|------|--------|------|--------|------|--------|------|
| GETCAN | 0007 | GETCHR | 0007 | GETREC | 0005 | GETSEC | F39D |
| GLHAB | 02E0 | GOOD | 0001 | GPRIOR | 026F | GRACIL | D01D |
| GRABN | D011 | GRAFFO | D00D | GRAFP1 | D00E | GRAFP2 | D00F |
| GRABP1 | D010 | HARD1 | F281 | HATABS | 031A | HITCLR | D01E |
| HOLD1 | 0051 | HOLD2 | 029F | HOLD3 | 029D | HOLD4 | 028C |
| HOLD5 | 028D | HOLDCH | 007C | HQWCH | F263 | HPOSMO | D004 |
| HPOSM1 | D005 | HPOSM2 | D006 | HPOSM3 | D007 | HPOSP0 | D000 |
| HPOSP1 | D001 | HPOSP2 | D002 | HPOSP3 | D003 | HSCROL | D404 |
| ICAX1 | 034A | ICAX1Z | 002A | ICAX2 | 034B | ICAX2Z | 002B |
| ICBAH | 0345 | ICBAHZ | 0025 | ICBAL | 0344 | ICBALZ | 0024 |
| ICBLH | 0349 | ICBLHZ | 0029 | ICBLL | 034B | ICBLLZ | 0028 |
| ICCOM | 0342 | ICCOMT | 0017 | ICCOMZ | 0022 | ICDNO | 0341 |
| ICDNI2 | 0021 | ICHID | 0340 | ICHIDZ | 0020 | ICIDNO | 002E |
| ICPT1 | 0347 | ICPTHZ | 0027 | ICPTL | 0346 | ICPTLZ | 0026 |
| ICSPR | 034C | ICSPRZ | 002C | ICSTA | 0343 | ICSTAZ | 0023 |
| IDENT | F0F2 | IDENTH | 00F0 | IDENTL | 00F2 | IFP | D9AA |
| INRUE | 00F3 | INIMLH | 0007 | INIMLL | 0000 | INSLR | 0020 |
| INODAT | 007D | INTABS | 0200 | INTEMP | 022D | INTINV | E46B |
| INTIRO | E4D5 | INTZBS | 0010 | INVFLQ | 0286 | IDCB | 0340 |
| IOCIAB | 0020 | IOCB8Z | 0010 | IOCFRE | 00FF | IRGEN | D20E |
| JRRT | D20E | KBCODE | D209 | KBD | 0048 | KBDORQ | F3E4 |
| KEYINDV | E420 | KEYDEL | 02F1 | LBFEND | 05FF | LBPR1 | 057E |
| LDIC | 057F | LBUFF | 0580 | LEDGE | 0002 | LINBUF | 0247 |
| LINZIB | 0000 | LIRQ | 0000 | LMARQN | 0032 | LOCKFL | 0023 |
| LIG | DECD | LQ010 | DED1 | LQ0COL | 0063 | LQ0MAP | 02B2 |
| LPENI | 0234 | LPENV | 0235 | MPFF | D000 | MPPL | D00B |
| NIPI | D001 | MIPL | D009 | M2PF | D002 | M2PL | D00A |
| NIPI | D003 | M3PL | D00B | MAXDEV | 0021 | MAXIOC | 0080 |
| NIPIU | 02E7 | MENTOP | 02E5 | MLTTP | 0066 | MODEM | 004D |
| NONIRQ | F0E3 | NONBR | 0014 | MOVVEC | F17D | MVBUFF | F32D |
| NONXII | F32F | MXDMOD | 0010 | NEWCOL | 0061 | NEWROM | 0060 |
| NONIEN | D40E | NMIREB | D40F | NMI8T | D40F | NOA1 | F1F1 |
| NOA2 | F212 | NOB1 | F1FB | NOBOOT | F1FF | NOCAR2 | F220 |
| NOCANT | F1FC | NOCKBM | 003C | NOC882 | F3BF | NOC8BT | F3E0 |
| NOIN11 | F2DC | NOKEY | F2CE | NONDEV | 0082 | NOTE | 0026 |
| NOTICPN | 00B5 | NOHARN | F2DD | NSION | 00EE | NVALID | 00B4 |
| NXTINT | F18C | OLDADR | 005E | OLDCHR | 005D | OLDCOL | 005B |
| OLDIIM | 005A | OPEN | 0003 | OPNEDT | F11B | OPNH | 00F1 |
| OPNIN | 0004 | OPNIND | 000C | OPNL | 0018 | OPNOT | 0008 |
| OPNTM | 0066 | OPBYB | F17B | OSRAM | F294 | OVRRUN | 008E |
| POPI | D004 | POPL | D00C | P1PF | D005 | P1PL | D00D |
| P2PF | D006 | P2PL | D00E | P3PF | D007 | P3PL | D00F |
| PACTL | D302 | PADDL0 | 0270 | PADDL1 | 0271 | PADDL2 | 0272 |
| PADDI3 | 0273 | PADDL4 | 0274 | PADDL5 | 0275 | PADDL6 | 0276 |
| PADDI7 | 0277 | PBCTL | D303 | PBPNT | 001D | PBUF8Z | 001E |
| PCOLRO | 02C0 | PCOLR1 | 02C1 | PCOLR2 | 02C2 | PCOLR3 | 02C3 |
| PENI | D40C | PENV | D40D | PIA | D300 | PLYARQ | 03E0 |
| PIYEV | D440 | PIBASE | D407 | POINT | 0025 | POKEY | D200 |
| POKMBK | 0010 | PORTA | D300 | PORTB | D301 | POT0 | D200 |
| POT1 | D201 | POT2 | D202 | POT3 | D203 | POT4 | D204 |
| POT5 | D205 | POT6 | D206 | POT7 | D207 | POT8 | D208 |
| PRINTR | 0050 | PRINTV | E430 | PRIOR | D01B | PRNBUF | 03C0 |
| PRNIRG | EE78 | PRVOPN | 0081 | PTEMP | 001F | PTIHOT | 001C |
| PTRIQ0 | 027C | PTRIQ1 | 027D | PTRIQ2 | 027E | PTRIQ3 | 027F |
| PTRIQ4 | 0280 | PTRIQ5 | 0281 | PTRIQ6 | 0282 | PTRIQ7 | 0283 |
| PUTCAR | 000B | PUTCHR | 000B | PUTLIN | F385 | PUTREC | 0009 |
| PUTXT | 0009 | PWRUP | F125 | PWRUP1 | F128 | RADFLQ | 00FB |
| RADIN | 0000 | RAMLO | 0004 | RAMSI2 | 02E4 | RAMTOP | 006A |
| RANDIM | D20A | RBLOKV | E47A | RDBYTE | F310 | RDONLY | 00B7 |
| READ | 0052 | RECVDN | 0039 | REDGE | 0027 | RENAME | 0020 |

| | | | | | | | |
|---------|------|---------|------|---------|------|--------|------|
| REFRT | F11B | RMARGN | 0053 | ROWAC | 0070 | ROWCRS | 0054 |
| ROWING | 0079 | RTCLCK | 0012 | SAVADR | 0068 | SAVID | 0316 |
| SAVADR | 0058 | SCREDT | 0045 | SCRENV | E410 | SCRFL0 | 0288 |
| SCRNM | 0093 | SCRNOK | F1DB | SDLSTH | 0231 | SDLSTL | 0230 |
| SDMCTI | 022F | SECT1 | F301 | SECTX | F34C | SENDEV | E468 |
| SERIN | D20D | SEROUT | D20D | SETVBV | E45C | SEX | 0000 |
| SIFANT | 006F | SHFLCK | 028E | SIGNON | F223 | SIN | BD81 |
| SIOINV | E463 | SIOOR0 | E944 | SIOV | E459 | SIZEM | D00C |
| SIZIPO | D008 | SIZEP1 | D009 | SIZEP2 | D00A | SIZEP3 | D00B |
| SKCTH | D20F | SKRES | D20A | SKSTAT | D20F | BOUNDR | 0041 |
| SLCCL | 000E | SPECL | F23F | SQR | BE01 | SRTIMR | 022B |
| SRFLA0 | 02FF | SRKCTL | 0232 | STACKP | 0318 | STATC | 0053 |
| STAT1B | 000D | STATU8 | 0030 | BTICK0 | 027B | BTICK1 | 0279 |
| BTICK2 | 027A | BTICK3 | 027B | BTIMER | D209 | STRIG0 | 0284 |
| STRIG1 | 0285 | STRIG2 | 0286 | STRIG3 | 0287 | SUBTMP | 029E |
| SUCCESS | 0001 | SWPFL0 | 007B | SVSVBV | E45F | TABMAP | 02A3 |
| TIDLEN | F0E3 | TIDLEN | 000E | TEMP | 023E | TEMP1 | 0312 |
| TEMP2 | 0314 | TEMP3 | 0315 | TIMER1 | 030C | TIMER2 | 0310 |
| TIMFL0 | 0317 | TIMOUT | 008A | TINDEX | 0293 | TMPCHR | 0050 |
| TIMCOL | 0289 | TIMPLBT | 02A1 | TIMPROW | 0288 | TMPX1 | 029C |
| TRANC2 | 0006 | TRIG0 | D010 | TRIG1 | D011 | TRIG2 | D012 |
| TRIG3 | D013 | TRNRCD | 0089 | TSTAT | 0319 | T8TDAT | 0007 |
| TXTCN | 0291 | TXTM8C | 0294 | TXTOLD | 0296 | TXTR0W | 0290 |
| UNLOCK | 0024 | USAREA | D480 | VBREAK | 0206 | VCOUNT | D40B |
| VCTANL | E480 | VDELAY | D01C | VD8LBT | 0200 | VECTBL | E400 |
| VIMIR0 | 0216 | VINTER | 0204 | VKEYBD | 0208 | VPRCED | 0202 |
| VSCRIN | D405 | VSERIN | 020A | VSEYDC | 020E | VSEYDC | 020C |
| VTIMR1 | 0210 | VTIMR2 | 0212 | VTIMR4 | 0214 | VVBLKD | 0224 |
| VWBIKI | 0222 | WARMST | 000B | WARMSV | E474 | WMODE | 0289 |
| WRITE | 00B7 | WROLY | 00B3 | WSYNC | D40A | XBOOT | F361 |
| X11VVV | E462 | XMTDON | 003A | ZERORH | F13B | Z10CB | 0020 |
| ZERRAM | F160 | ZOBRM2 | F163 | ZOBRM3 | F16E | ZTEMP1 | 00F5 |
| ZTEMP3 | 00F9 | ZTEMP4 | 00F7 | | | | |

DISPLAY HANDLER -- 10-30-78 -- DISPLC

TITLE 'DISPLAY HANDLER -- 10-30-78 -- DISPLC'

HANDLER DEPENDENT EQUATES

| | | | | |
|--------|--------|---|--------|---------------------------|
| (007B) | CR | = | \$9B | SYSTEM EOL |
| (007D) | CLRCOD | = | \$7D | CLEAR SCREEN ATASCII CODE |
| (009F) | CNTLI | = | \$9F | POKEY KEY CODE FOR ^I |
| (006B) | FRMADR | = | SAVADR | |
| (0066) | TOADR | = | MLTTMP | |

==EDITRV

SCREEN EDITOR HANDLER ENTRY POINT

| | | | | |
|---------------|--------|------|----------|-----------------|
| F400 FB F3 | EDITOR | WORD | EOPEN-1 | |
| F402 33 F6 | | WORD | RETUR1-1 | (CLOSE) |
| F404 3D F6 | | WORD | EGETCH-1 | |
| F406 A3 F6 | | WORD | EOUTCH-1 | |
| F408 33 F6 | | WORD | RETUR1-1 | (STATUS) |
| F40A 3C F6 | | WORD | NOFUNC-1 | (SPECIAL) |
| F40C 4C E4 F3 | | JMP | PWRONA | |
| F40F 00 | | BYTE | 0 | ROM FILLER BYTE |

==SCRENV

DISPLAY HANDLER ENTRY POINT

| | | | | |
|---------------|--------|------|----------|-----------------|
| E410 F5 F3 | DISPLA | WORD | DOPEN-1 | |
| E412 33 F6 | | WORD | RETUR1-1 | (CLOSE) |
| E414 92 F5 | | WORD | OETCH-1 | |
| E416 B6 F5 | | WORD | OUTCH-1 | |
| E418 33 F6 | | WORD | RETUR1-1 | (STATUS) |
| E41A FB F6 | | WORD | DRAW-1 | (SPECIAL) |
| E41C 4C E4 F3 | | JMP | PWRONA | |
| E41F 00 | | BYTE | 0 | ROM FILLER BYTE |

==KEYBDV

KEYBOARD HANDLER ENTRY POINT

| | | | | |
|---------------|--------|------|----------|-----------------|
| F420 33 F6 | KBD#ND | WORD | RETUR1-1 | |
| F422 33 F6 | | WORD | RETUR1-1 | (CLOSE) |
| E424 E1 F6 | | WORD | KOETCH-1 | |
| E426 3C F6 | | WORD | NOFUNC-1 | (DUTCH) |
| E428 33 F6 | | WORD | RETUR1-1 | (STATUS) |
| F42A 3C F6 | | WORD | NOFUNC-1 | (SPECIAL) |
| F42C 4C E4 F3 | | JMP | PWRONA | |
| F42F 00 | | BYTE | 0 | ROM FILLER BYTE |

INTERRUPT VECTOR TABLE ENTRY

| | | | | |
|-------------|------------------------|------|------|-------------------------------|
| I:488 BE FF | ==VCTABL-INTABS+VKEYBD | WORD | PIRQ | KEYBOARD IRQ INTERRUPT VECTOR |
|-------------|------------------------|------|------|-------------------------------|

**KBDORO

| | | | | |
|---------------|--------|-----|----------|--------------------------------------|
| 13E4 A9 FF | PWRONA | LDA | #FF | |
| 13E6 BD FC 02 | | STA | CH | |
| 13E9 AD E6 02 | | LDA | MENTOP+1 | |
| 13EC 29 F0 | | AND | #F0 | INSURE 4K PAGE BOUNDARY |
| 13EE B5 6A | | STA | RAMTOP | |
| 13F0 A9 40 | | LDA | #40 | DEFAULT TO UPPER CASE ALPHA AT PWRON |
| 13F2 BD BE 02 | | STA | SHFLOK | |
| 13F5 60 | | RTB | | POWER ON COMPLETED |

```

      BEGIN DISPLAY HANDLER OPEN PROCESSING
I3F6 A5 2B      DOPEN LDA ICAX2Z GET AUX 2 BYTE
I3FB 29 0F      AND    #0F
I3FA D0 0B      BNE    OPNCOM IF MODE ZERO, CLEAR ICAX1Z
I3FC A5 2A      EOPEN LDA ICAX1Z CLEAR "CLR INHIBIT" AND "H&D MODE" BITS
I3FE 29 0F      AND    #0F
I400 B5 2A      STA    ICAX1Z
I402 A7 00      LDA    #0
F404 B5 57      OPNCOM STA DINDEX
I40A A9 E0      LDA    #0E0 INITIALIZE GLOBAL VBLANK RAM
F40B BD F4 02   STA    CHDAB
I40B A9 02      LDA    #2
I40D BD F3 02   STA    CHACT
I410 BD 2F 02   STA    SDMCTL TURN OFF DMA NEXT VBLANK
F413 A9 01      LDA    #SUCCESS
F41B B9 AC      STA    DBSTAT CLEAR STATUS
F417 A7 C0      LDA    #C0 DO IRGEN
I419 05 10      ORA    POKMSK
I41B B5 10      STA    POKMSK
I41D BD 0E D2   STA    IRGEN
F420 A9 00      LDA    #0
I422 BD 93 02   STA    TINDEX TEXT INDEX MUST ALWAYS BE 0
I425 B5 64      STA    ADDRESS
I427 B5 7B      STA    SWPFL0
I429 BD F0 02   STA    CRSINH TURN CURSOR ON AT OPEN
I42C A0 0E      LDY    #14 CLEAR TAB STOPS
F42E A9 D1      LDA    #1 INIT TAB STOPS TO EVERY 8 CHARACTERS
F430 99 A3 02   CLRTBS STA TABMAP
F433 B8        DEY
I434 10 FA      BPL    CLRTBS
I436 A2 04      LDX    #4 LOAD COLOR REGISTERS
I43B BD C1 FE   DOPENB LDA COLRTB,X
F43B 9D C4 02   STA    COLOR,X
F43E CA        DEX
F43F 10 F7      BPL    DOPENB
I441 A4 6A      LDY    RANTOP DO TXTHSC=$2C40 (IF MEMTOP=3000)
I443 B8        DEY
I444 BC 95 02   STY    TXTHSC+1
F447 A9 60      LDA    #0
F449 BD 94 02   STA    TXTHSC
F44C A6 57      LDX    DINDEX
I44E BD 69 FE   LDA    ANCONV,X CONVERT IT TO ANTIC CODE
I451 D0 04      BNE    DOPENA IF ZERO, IT IS ILLEGAL
F453 A9 91      OPNERR LDA #BADMOD SET ERROR STATUS
F455 B5 4C      STA    DBSTAT
F457 B5 51      DOPENA STA HOLD1
F459 A5 6A      LDA    RANTOP SET UP AN INDIRECT POINTER
I45B B5 65      STA    ADDRESS+1
I45D BC 45 FE   LDY    ALOCAT,X ALLOCATE N BLOCKS OF 40 BYTES
I460 A9 2B      DOPENI LDA #40
F462 20 21 F9   JSR    DBSUB
F465 B8        DEY
F466 D0 F8      BNE    DOPENI
I46B AD 6F 02   LDA    OPRIOR CLEAR OTIA MODES
I46B 29 3F      AND    #03F
I46D B5 67      STA    OPNTMP+1
F46F AB        TAY

```

| | | | |
|----------------|-----|-----------|---|
| I-470 E0 08 | CPX | #8 | TEST IF 320X1 |
| I-472 90 17 | BCC | NOT8 | |
| I-474 8A | TXA | | GET 2 LOW BITS |
| I-475 6A | ROR | A | |
| I-476 6A | ROR | A | |
| I-477 6A | ROR | A | |
| I-478 29 C0 | AND | #6C0 | NOW 2 TOP BITS |
| I-47A 05 67 | ORA | OPNTMP+1 | |
| I-47C AB | TAY | | |
| I-47D A9 10 | LDA | #16 | SUBTRACT 16 MORE FOR PAGE BOUNDARY |
| I-47F 20 21 F9 | JSR | DBSUB | |
| F-482 E0 08 | CPX | #11 | TEST MODE 11 |
| F-484 D0 05 | BNE | NOT8 | IF MODE = 11 |
| F-486 A9 06 | LDA | #6 | PUT OTIA LUM VALUE INTO BACKGROUND REGISTER |
| F-488 BD C8 02 | STA | COLOR4 | |
| F-48B BC 6F 02 | STY | OPRIOR | STORE NEW PRIORITY |
| F-48E A5 64 | LDA | ADRE88 | SAVE MEMORY SCAN COUNTER ADDRESS |
| F-490 B5 58 | STA | SAVMBC | |
| F-492 A5 65 | LDA | ADRE88+1 | |
| F-494 B5 59 | STA | SAVMBC+1 | |
| F-496 AD 08 D4 | LDA | VCOUNT | WAIT FOR NEXT VBLANK BEFORE MESSING |
| F-499 C9 7A | CMF | #7A | WITH THE DISPLAY LIST |
| F-49B D0 F9 | BNE | VBWAIT | |
| F-49D 20 1F F9 | JSR | DBDEC | START PUTTING DISPLAY LIST RIGHT UNDER RAM |
| F-4A0 BD 78 FE | LDA | PAGETB, K | TEST IF DISPLAY LIST WILL BE IN TROUBLE |
| F-4A3 F0 06 | BEG | NOMOD | OF CROSSING A 256 BYTE PAGE BOUNDARY |
| F-4A5 A9 FF | LDA | #FF | IF 60, DROP DOWN A PAGE |
| F-4A7 B5 64 | STA | ADRE88 | |
| F-4A9 C6 65 | DEC | ADRE88+1 | |
| F-4AB A5 64 | LDA | ADRE88 | SAVE END OF DISPLAY LIST FOR LATER |
| F-4AD B5 68 | STA | SAVADR | |
| F-4AF A5 65 | LDA | ADRE88+1 | |
| F-4B1 B5 69 | STA | SAVADR+1 | |
| F-4B3 20 13 F9 | JSR | DBDEC | (DOUBLE BYTE DOUBLE DECREMENT) |
| F-4B6 A9 41 | LDA | #41 | (ANTIC) WAIT FOR VBLANK AND JMP TO TOP |
| F-4BB 20 17 F9 | JSR | STORE | |
| F-4BB B6 66 | STX | OPNTMP | |
| F-4BD A9 18 | LDA | #24 | INITIALIZE BOTSCR |
| F-4BF BD BF 02 | STA | BOTSCR | |
| F-4C2 A3 57 | LDA | DINDEX | DISALLOW MIXED MODE IF MODE. 0E. 9 |
| F-4C4 C9 09 | CMF | #9 | |
| F-4C6 B0 2D | BCC | NOTMXD | |
| F-4C8 A3 2A | LDA | ICAX1Z | TEST MIXED MODE |
| F-4CA 29 10 | AND | #10 | |
| F-4CC F0 27 | BEG | NOTMXD | |
| F-4CE A9 04 | LDA | #4 | |
| F-4D0 BD BF 02 | STA | BOTSCR | |
| F-4D3 A2 02 | LDY | #2 | ADD 4 LINES OF TEXT AT BOTTOM OF SCREEN |
| F-4D5 A9 02 | LDA | #2 | |
| F-4D7 20 17 F9 | JSR | STORE | |
| F-4DA CA | DEX | | |
| F-4DB 10 F8 | BPL | DOPEN2 | |
| F-4DD A4 6A | LDY | RANTOP | RELOAD MBC FOR TEXT |
| F-4DF B8 | DEY | | |
| F-4E0 98 | TYA | | |
| F-4E1 20 17 F9 | JSR | STORE | |
| F-4E4 A9 60 | LDA | #60 | |
| F-4E6 20 17 F9 | JSR | STORE | |
| F-4E9 A9 42 | LDA | #42 | |
| F-4EB 20 17 F9 | JSR | STORE | |

| | | |
|---------------|-----|--|
| F4EE 18 | CLC | |
| F4EF A9 0C | LDA | #MXDMDE-MUMBLE POINT X AT MIXED MODE TABLE |
| F4F1 65 66 | ADC | OPNTMP |
| F4F3 85 66 | BTA | OPNTMP |
| F4F5 A4 66 | LDY | OPNTMP |
| F4F7 8E 81 FE | LDX | NUMBLE.Y GET NUMBER OF DISPLAY LIST ENTRIES |
| F4FA A5 91 | LDA | HOLD1 STORE N DLE'S |
| F4FC 20 17 F9 | JBR | STORE |
| F4FF CA | DEX | |
| F500 D0 FB | BNE | DOPEN3 |
| F502 A5 57 | LDA | RINDEX DO THE MESSY 320X1 PROBLEM |
| F504 C9 0B | CMP | #B |
| F506 90 1C | BCC | DOPEN5 |
| F508 A2 5D | LDX | #93 GET REMAINING NUMBER OF DLE'S |
| F50A A5 6A | LDA | RANTOP RELOAD MEMORY SCAN COUNTER |
| F50C 3B | BEC | |
| F50D E9 10 | SBC | #10 |
| F50F 20 17 F9 | JBR | STORE |
| F512 A9 00 | LDA | #0 |
| F514 20 17 F9 | JBR | STORE |
| F517 A9 4F | LDA | #4F (ANTIC) RELOAD MSC CODE |
| F519 20 17 F9 | JBR | STORE |
| F51C A5 51 | LDA | HOLD1 DO REMAINING DLE'S |
| F51E 20 17 F9 | JBR | STORE |
| F521 CA | DEX | |
| F522 D0 FB | BNE | DOPEN4 |
| F524 A5 59 | LDA | SAVMSC+1 POLISH OFF DISPLAY LIST |
| F526 20 17 F9 | JBR | STORE |
| F529 A5 5B | LDA | SAVMSC |
| F52B 20 17 F9 | JBR | STORE |
| F52E A5 91 | LDA | HOLD1 |
| F530 09 40 | ORA | #40 |
| F532 20 17 F9 | JBR | STORE |
| F535 A9 70 | LDA | #70 24 BLANK LINES |
| F537 20 17 F9 | JBR | STORE |
| F53A A9 70 | LDA | #70 |
| F53C 20 17 F9 | JBR | STORE |
| F53F A5 64 | LDA | ADRESS SAVE DISPLAY LIST ADDRESS |
| F541 8D 30 02 | BTA | BDLSTL |
| F544 A5 65 | LDA | ADRESS+1 |
| F546 8D 31 02 | BTA | BDLSTL+1 |
| F549 A9 70 | LDA | #70 ADD LAST BLANK LINE ENTRY |
| F54B 20 17 F9 | JBR | STORE POSITION ADRESS-BDLSTL-1 |
| F54E A5 64 | LDA | ADRESS STORE NEW MEMTOP |
| F550 8D E5 02 | BTA | MEMTOP |
| F553 A5 65 | LDA | ADRESS+1 |
| F555 8D E6 02 | BTA | MEMTOP+1 |
| F558 A5 68 | LDA | SAVADR |
| F55A 85 64 | BTA | ADRESS |
| F55C A5 69 | LDA | SAVADR+1 |
| F55E 85 65 | BTA | ADRESS+1 |
| F560 AD 31 02 | LDA | BDLSTL+1 |
| F563 20 17 F9 | JBR | STORE |
| F566 AD 30 02 | LDA | BDLSTL |
| F569 20 17 F9 | JBR | STORE |
| F56C A5 4C | LDA | DBTAT IF ERROR OCCURRED ON ALLOCATION, OPEN THE EDITOR |
| F56E 10 07 | BPL | DOPEN9 |
| F570 4B | PHA | SAVE STATUS |
| F571 20 FC F3 | JBR | EOPEN OPEN THE EDITOR |
| F574 6B | PLA | RESTORE STATUS |


```

1575 AB          TAY          AND RETURN IT TO CIO
1576 60          RTS
1577 A5 2A      DOPEN9 LDA    ICAX1Z TEST CLEAR INHIBIT BIT
1579 29 20      AND      #620
157B D0 0B      BNE      DOPEN7
157D 20 B9 F7   JSR      CLRBCR CLEAR SCREEN
1580 8D 90 02   STA      TXTROW AND HOME TEXT CURSOR (AC IS ZERO)
1583 A5 52      LDA      LMARON
1585 8D 91 02   STA      TXTCOL
1588 A9 22      DOPEN7 LDA    #622 EVERYTHING ELSE IS SET UP
158A 0D 2F 02   DRA      SDMCTL SO TURN ON DMACTL
158D 8D 2F 02   STA      SDMCTL
1590 4C 21 F4   JMP      RETURN
  
```

```

1593 20 96 FA   GETCH JSR    RANGE GETCH DOES INCRBR, GETPLT DOESN'T
1596 20 A2 F5   JSR    GETPLT
1599 20 32 F8   JSR    INATAC CONVERT INTERNAL CODE TO ATASCII
159C 20 D4 F9   JSR    INCRBR
159F 4C 34 F6   JMP     RETURN
15A2 20 47 F9   GETPLT JSR    CONVRT CONVERT ROW/COLUMN TO ADRESS
15A5 B1 64      LDA      (ADRESS),Y
15A7 2D A0 02   AND      DMASK
15AA 44 6F      SHIFTD LSR    SHIFTD SHIFT DATA DOWN TO LOW BITS
15AC 8D 03      RCB      SHIFTD
15AE 4A         LSR      A
15AF 10 F9      DPL      SHIFTD (UNCONDITIONAL)
15B1 8D FA 02   SHIFTD STA    CHAR
15B4 C9 00      CMP      #0 RESTORE FLAG8 ALSO
  
```

```

15B7 8D FB 02   OUTCH STA    ATACHR
15BA 20 96 FA   JSR    RANGE
15BD AD FB 02   OUTCHA LDA    OFFCRB
15C0 C9 7D      CMP      ATACHR TEST FOR CLEAR SCREEN
15C2 D0 06      BNE      #CLRCOD
15C4 20 B9 F7   JSR      OUTCHE
15C7 4C 21 F6   JMP      CLRBCR
15CA AD FB 02   OUTCHE LDA    ATACHR TEST FOR CARRIAGE RETURN
15CD C9 98      CMP      #CR
15CF D0 06      BNE      OUTCHB
15D1 20 30 FA   JSR      DOCRWB DO CR
15D4 4C 21 F6   JMP      RETURN
15D7 20 E0 F5   OUTCHB JSR    OUTPLT
15DA 20 D8 F9   JSR      INCRBR
15DD 4C 21 F6   JMP      RETURN
  
```

```

15E0 AD FF 02   OUTPLT LDA    SBFLAG *****LOOP HERE IF START/STOP FLAG IS NON-0
15E3 D0 FB      BNE      OUTPLT
15E5 A2 02      LDX      #2
15E7 B9 54      CRLOOP LDA    ROWCRB, X SAVE CURSOR LOCATION FOR DRAW LINE TO DRAW FROM
15E9 95 5A      STA      OLDROW, X
15EB CA         DEX
15EC 10 F9      BPL      CRLOOP
15EE AD FB 02   LDA      ATACHR CONVERT ATASCII(ATACHR) TO INTERNAL(CHAR)
15F1 AB          TAY      SAVE ATACHR
  
```


| | | | | |
|----------------|--------|-----|-----|---|
| I-5F2 2A | | ROL | A | |
| I-5F3 2A | | ROL | A | |
| I-5F4 2A | | ROL | A | |
| I-5F5 2A | | ROL | A | |
| I-5F6 29 03 | | AND | #3 | |
| I-5F8 AA | | TAX | | X HAS INDEX INTO ATAIN |
| I-5F9 9B | | TYA | | RESTORE ATACHR |
| I-5FA 29 9F | | AND | #9F | STRIP OFF COLUMN ADDRESS |
| I-5FC 1D F6 FE | | ORA | | ATAINT, X OR IN NEW COLUMN ADDRESS |
| I-5FF 8D FA 02 | OUTCH2 | STA | | CHAR |
| I-602 20 47 F9 | | JSR | | CONVRT |
| I-605 AD FA 02 | | LDA | | CHAR |
| I-608 46 6F | SHIFTU | LBR | | SHFAMT SHIFT UP TO PROPER POSITION |
| I-60A 80 04 | | BCS | | SHIFTE |
| I-60C 0A | | ASL | A | |
| I-60D 4C 0B F6 | | JMP | | SHIFTU |
| I-610 2D A0 02 | SHIFTE | AND | | DMASK |
| I-613 85 50 | | STA | | TMPCHR SAVE SHIFTE DATA |
| I-615 AD A0 02 | | LDA | | DMASK INVERT MASK |
| I-618 49 FF | | EOR | | #FF |
| I-61A 31 64 | | AND | | (ADDRESS), Y MASK OFF OLD DATA |
| I-61C 05 50 | | ORA | | TMPCHR OR IN NEW DATA |
| I-61E 91 64 | | STA | | (ADDRESS), Y |
| I-620 60 | | RTB | | |
| I-621 20 A2 F5 | RETURN | JSR | | GETPLY DO CURSOR ON THE WAY OUT |
| I-624 85 5D | | STA | | OLDCHR |
| I-626 A6 57 | | LDX | | DINDEX GRAPHICS HAVE INVISIBLE CURSOR |
| I-628 D0 0A | | BNE | | RETURN |
| I-62A AE F0 02 | | LDX | | CRSINH TEST CURSOR INHIBIT |
| I-62D D0 05 | | BNE | | RETURN |
| I-62F 49 80 | | EOR | | #80 TOGGLE MSB |
| I-631 20 FF F5 | | JSR | | OUTCH2 DISPLAY IT |
| I-634 A4 4C | RETURN | LDY | | DSTAT RETURN TO CIO WITH STATUS IN Y |
| I-636 A9 01 | | LDA | | #SUCCESS |
| I-638 85 4C | | STA | | DSTAT SET STATUS SUCCESSFUL COMPLETION |
| I-63A AD FB 02 | | LDA | | ATACHR PUT ATACHR IN AC FOR RETURN TO CIO |
| I-63D 60 | NOFUNC | RTB | | (NON-EXISTENT FUNCTION RETURN POINT) |

END OF DISPLAY HANDLER

| | | | | |
|---------------|--------|-----|-----------|--|
| 163E 20 B3 FC | EOETCH | JSR | SWAP | |
| 1641 20 B8 FA | | JSR | ERANGE | |
| 1644 A3 68 | | LDA | BUFCNT | ANYTHING IN THE BUFFER? |
| 1646 D0 34 | | BNE | EOETC3 | YES |
| 1648 A5 54 | | LDA | ROWCRS | NO, SO SAVE BUFFER START ADDRESS |
| 164A B5 6C | | STA | BUFSTR | |
| 164C A5 55 | | LDA | COLCRS | |
| 164E B5 6D | | STA | BUFSTR+1 | |
| 1650 20 E2 F6 | EOETC1 | JSR | EOETCH | LET'S FILL OUR BUFFER |
| 1653 B4 4C | | STY | DSTAT | SAVE KEYBOARD STATUS |
| 1655 AD FB 02 | | LDA | ATACHR | TEST FOR CR |
| 1658 C9 9B | | CMF | #CR | |
| 165A F0 12 | | BEQ | EOETC2 | |
| 165C 20 AD F6 | | JSR | DOBS | NO, SO PRINT IT |
| 165F 20 B3 FC | | JSR | SWAP | JSR DOBS DID SWAP SO SWAP BACK |
| 1662 A5 63 | | LDA | LOGCOL | DEEP IF NEARING LOGICAL COL 120 |
| 1664 C9 71 | | CMF | #113 | |
| 1666 D0 03 | | BNE | EOETC6 | |
| 1668 20 0A F9 | | JSR | BELL | |
| 166B 4C 50 F4 | EOETC4 | JMP | EOETC1 | |
| 166E 20 E4 FA | EOETC2 | JSR | OFFCRS | GET BUFFER COUNT |
| 1671 20 00 FC | | JSR | DOBUFC | |
| 1674 A5 6C | | LDA | BUFSTR | RETURN A CHARACTER |
| 1676 B5 54 | | STA | ROWCRS | |
| 1678 A5 6D | | LDA | BUFSTR+1 | |
| 167A B5 55 | | STA | COLCRS | |
| 167C A5 68 | EOETC3 | LDA | BUFCNT | |
| 167E F0 11 | | BEQ | EOETC5 | |
| 1680 C6 6B | EOETC7 | DEC | BUFCNT | AND RETURN TILL BUFCNT=0 |
| 1682 F0 0D | | BEQ | EOETC5 | |
| 1684 A5 4C | | LDA | DSTAT | IF ERROR, LOOP ON EOETC7 UNTIL BUFFER IS EMPTIED |
| 1686 30 FB | | BHI | EOETC7 | |
| 1688 20 93 F5 | | JSR | EOETCH | |
| 168B BD FB 02 | | STA | ATACHR | |
| 168E 4C B3 FC | | JMP | SWAP | AND RETURN WITHOUT TURNING CURSOR BACK ON |
| 1691 20 30 FA | EOETC5 | JSR | DOCRWS | DO REAL CARRIAGE RETURN |
| 1694 A9 9B | | LDA | #CR | AND RETURN EOL |
| 1696 BD FB 02 | | STA | ATACHR | |
| 1699 20 21 F6 | | JSR | RETURN | TURN ON CURSOR THEN SWAP |
| 169C B4 4C | | STY | DSTAT | SAVE KEYBOARD STATUS |
| 169E 4C B3 FC | | JMP | SWAP | AND RETURN THROUGH RETUR1 |
| 16A1 6C 64 00 | JSRIND | JMP | (ADDRESS) | JSR TO THIS CAUSES JSR INDIRECT |
| 16A4 BD FB 02 | EOETCH | STA | ATACHR | SAVE ATACHR VALUE |
| 16A7 20 B3 FC | | JSR | SWAP | |
| 16AA 20 B8 FA | | JSR | ERANGE | |
| 16AD 20 E4 FA | DOBS | JSR | OFFCRS | TURN OFF CURSOR |
| 16B0 20 BD FC | | JSR | TSCTCL | TEST FOR CONTROL CHARACTERS (Z=1 IF CTL) |
| 16B3 F0 09 | | BEQ | EOETC5 | |
| 16B5 0E A2 D2 | EOETC6 | ASL | ESCFLO | ESCFLO ONLY WORKS ONCE |
| 16B8 20 CA F5 | | JSR | EOETCHE | |
| 16BB 4C B3 FC | ERETN | JMP | SWAP | AND RETURN THROUGH RETUR1 |
| 16BE AD FE 02 | EOETC5 | LDA | DSPFLO | DO DSPFLO AND ESCFLO |
| 16C1 0D A2 02 | | ORA | ESCFLO | |
| 16C4 D0 EF | | BNE | EOETC6 | IF NON-0 DISPLAY RATHER THAN EXECUTE IT |

| | | | |
|---------------|-----|------------|-----------------------------------|
| F6C6 0E A2 02 | ASL | EBCFL0 | |
| F6C9 E8 | INX | | PROCESS CONTROL CHARACTER |
| F6CA BD C6 FE | LDA | CNTRLB,X | GET DISPLACEMENT INTO ROUTINE |
| F6CD 85 64 | STA | ADDRESS | |
| F6CF BD C7 FE | LDA | CNTRLB+1,X | GET HIGH BYTE |
| F6D2 B5 63 | STA | ADDRESS+1 | |
| F6D4 20 A1 F4 | JBR | JBRIND | DO COMPUTED JBR |
| F6D7 20 21 F6 | JBR | RETURN | DO CURSOR |
| F6DA 4C B3 FC | JMP | SWAP | ALL DONE SO RETURN THROUGH RETURN |

END SCREEN EDITOR

BEGIN KEYBOARD HANDLER

| | | | | |
|---------------|--------|-----|----------|--|
| F6DD A9 FF | KOETC2 | LDA | #FF | |
| F6DF BD FC 02 | | STA | CH | |
| F6E2 A5 2A | KOETCH | LDA | ICANIZ | TEST LBB OF AUX1 FOR SPECIAL EDITOR READ MODE |
| F6E4 4A | | LBR | A | |
| F6E8 B0 62 | | BCH | GETONT | |
| F6E7 A9 80 | | LDA | #BRKABT | |
| F6E9 A6 11 | | LDX | BRKKEY | TEST BREAK |
| F6EB F0 58 | | BEQ | K7 | IF BREAK, PUT BRKABT IN DSTAT AND CR IN ATACHR |
| F6ED AD FC 02 | | LDA | CH | |
| F6F0 C9 FF | | CMR | #FF | |
| F6F2 F0 EE | | BEQ | KOETCH | |
| F6F4 B5 7C | | STA | HOLDCH | SAVE CH FOR SHIFT LOCK PROC |
| F6F6 A2 FF | | LDX | #FF | "CLEAR" CH |
| F6FB 8E FC 02 | | STX | CH | |
| F6FB 20 DB FC | | JBR | CLICK | DO KEYBOARD AUDIO FEEDBACK (A IS OK) |
| F6FE AA | KOETC3 | TAX | | DO ABCCON |
| F6FF E0 C0 | | CPX | #C0 | TEST FOR CTL & SHIFT TOGETHER |
| F701 90 02 | | BCC | ABCC01 | |
| F703 A2 03 | | LDX | #3 | BAD CODE |
| F705 BD FE FE | ABCC01 | LDA | ATABCI,X | |
| F708 BD FB 02 | | STA | ATACHR | DONE |
| F708 C9 80 | | CMR | #80 | DO NULL |
| F70D F0 CE | | BEQ | KOETC2 | |
| F70F C9 81 | | CMR | #81 | CHECK ATARI KEY |
| F711 D0 0B | | BNE | KOETC1 | |
| F713 AD B6 02 | | LDA | INVFL0 | |
| F71A 49 B0 | | EOR | #80 | |
| F71B BD B6 02 | | STA | INVFL0 | |
| F71B 4C DD F6 | | JMP | KOETC2 | DONT RETURN A VALUE |
| F71E C9 82 | KOETC1 | CMR | #82 | CAPS/LOWER |
| F720 D0 07 | | BNE | K1 | |
| F722 A9 00 | | LDA | #0 | CLEAR SHFLOK |
| F724 BD BE 02 | | STA | SHFLOK | |
| F727 F0 84 | | BEQ | KOETC2 | |
| F729 C9 83 | K1 | CMR | #83 | SHIFT CAPS/LOWER |
| F72B D0 07 | | BNE | K2 | |
| F72D A9 40 | | LDA | #40 | |
| F72F BD DE 02 | | STA | SHFLOK | SHIFT BIT |
| F732 D0 A9 | | BNE | KOETC2 | |

| | | | | |
|---------------|--------|-----|---------|---|
| 1734 C9 84 | K2 | CMP | #B4 | CNTL CAPS/LOWER |
| 1736 D0 07 | | BNE | K3 | |
| 1738 A9 80 | | LDA | #B0 | CNTL BIT |
| 173A 8D 0E 02 | | STA | SHFLOK | |
| 173D D0 7E | | BNE | K0ETC2 | |
| 173F C9 85 | K3 | CMP | #B5 | DO EOF |
| 1741 D0 0A | | BNE | K6 | |
| 1743 A9 88 | | LDA | NEOFERR | |
| 1745 85 4C | K7 | STA | DSTAT | |
| 1747 85 11 | | STA | BRKKEY | RESTORE BREAK |
| 1749 A9 9B | GETOUT | LDA | WCR | PUT CR IN ATACHR |
| 174B D0 26 | | BNE | K8 | (UNCONDITIONAL) |
| 174D A5 7C | K6 | LDA | HOLDCH | PROCESS SHIFT LOCKS |
| 174F C9 40 | | CMP | #40 | REGULAR SHIFT AND CONTROL TAKE PRECEDENCE |
| 1751 B0 15 | | BCB | K5 | OVER LOCK |
| 1753 AD FB 02 | | LDA | ATACHR | TEST FOR ALPHA |
| 1756 C9 61 | | CMP | #61 | LOWER CASE A |
| 1758 90 0E | | BCC | K5 | NOT ALPHA IF LT |
| 175A C9 7B | | CMP | #7B | LOWER CASE Z+1 |
| 175C B0 0A | | BCB | K5 | NOT ALPHA IF GE |
| 175E AD BE 02 | | LDA | SHFLOK | DO SHIFT/CONTROL LOCK |
| 1761 F0 05 | | BEG | K5 | IF NO LOCK, DONT RE-DO IT |
| 1763 05 7C | | ORA | HOLDCH | |
| 1765 4C FE F6 | | JMP | K0ETC3 | DO RETRY |
| 1768 20 8D FC | K9 | JBR | T8TCTL | DONT INVERT MSB OF CONTROL CHARACTERS |
| 176B F0 09 | | BEG | K4 | |
| 176D AD FB 02 | | LDA | ATACHR | |
| 1770 4D 86 02 | | EDR | INVFL0 | |
| 1773 8D FB 02 | K8 | STA | ATACHR | |
| 1776 4C 34 F6 | K4 | JMP | RETURN | ALL DONE |

CONTROL CHARACTER PROCESSORS

| | | | | |
|---------------|--------|-----|---|---|
| 1779 A9 80 | ESCAPE | LDA | #90 | SET ESCAPE FLAG |
| 177B BD A2 02 | | STA | ESCFL0 | |
| 177E 60 | | RTB | | |
| 177F C6 54 | CRSRUP | DEC | ROWCRB | |
| 1781 10 06 | | BPL | COMRET | |
| 1783 AE BF 02 | | LDX | BOTSCR | WRAPAROUND |
| 1786 CA | | DEX | | |
| 1787 B6 54 | UPDNCH | STX | ROWCRB | |
| 1789 4C 5C FC | COMRET | JMP | STRBEQ | COLVERT ROW AND COL TO LOGCOL AND RETURN |
| 178C E6 54 | CRSRDN | INC | ROWCRB | |
| 178E A5 54 | | LDA | ROWCRB | |
| 1790 CD BF 02 | | CMP | BOTSCR | |
| 1793 90 F4 | | BCC | COMRET | |
| 1795 A2 00 | | LDX | #0 | |
| 1797 F0 EE | | BEQ | UPDNCH | (UNCONDITIONAL) |
| 1799 C6 55 | CRSRLE | DEC | COLCRB | |
| 179B A5 55 | | LDA | COLCRB | |
| 179D 30 04 | | BMI | CRSRL1 | (IF LMARON=0. THIS ELIMINATED PROBLEM CASE) |
| 179F C5 52 | | CMP | LMARON | |
| 17A1 B0 04 | | BCB | COMRE1 | |
| 17A3 A5 53 | CRSRL1 | LDA | RMARON | |
| 17A5 B5 53 | LFRTCM | STA | COLCRB | |
| 17A7 4C DD FB | COMRE1 | JMP | DOLCOL | COLVERT ROW AND COL TO LOGCOL AND RETURN |
| 17AA E6 55 | CRSRRT | INC | COLCRB | |
| 17AC A5 55 | | LDA | COLCRB | |
| 17AE C5 53 | | CMP | RMARON | |
| 17B0 90 F5 | | BCC | COMRE1 | |
| 17B2 F0 F3 | | BEQ | COMRE1 | (CAUSE BLE) |
| 17B4 A5 52 | | LDA | LMARON | |
| 17B6 4C A5 F7 | | JMP | LFRTCM | UNCONDITIONAL TO COMMON STORE |
| 17B9 20 F3 FC | CLRBCR | JSR | PUTMBC | |
| 17BC A0 00 | | LDY | #0 | |
| 17BE 98 | | TYA | | PUT 0 IN THE AC |
| 17BF 91 64 | CLRBC2 | STA | (ADDRESS), Y (AC IS ZERO) | |
| 17C1 C8 | | INY | | |
| 17C2 D0 F8 | | BNE | CLRBC2 | |
| 17C4 E6 65 | | INC | ADDRESS+1 | |
| 17C6 A6 65 | | LDX | ADDRESS+1 | |
| 17C8 E4 6A | | CPX | RAMTOP | |
| 17CA 90 F3 | | BCC | CLRBC2 | |
| 17CC A9 FF | | LDA | #FF | CLEAN UP LOGICAL LINE BIT MAP |
| 17CE 99 B2 02 | CLRBC3 | STA | LOOMAP, Y (Y IS ZERO AFTER CLRBC2 LOOP) | |
| 17D1 C8 | | INY | | |
| 17D2 C0 04 | | CPY | #4 | |
| 17D4 90 F8 | | BCC | CLRBC3 | |
| 17D6 20 E4 FC | HOME | JSR | COLCR | PLACE COLCRB AT LEFT EDGE |
| 17D9 B5 63 | | STA | LOGCOL | |
| 17DB B5 6D | | STA | BUFSTR+1 | |
| 17DD A9 00 | | LDA | #0 | |
| 17DF B5 54 | | STA | ROWCRB | |
| 17E1 B5 56 | | STA | COLCRB+1 | |
| 17E3 B5 6C | | STA | BUFSTR | |
| 17E5 60 | | RTS | | |

| | | | |
|---------------|--------|------------|--|
| 1873 A5 64 | LDA | ADRESS | |
| 1875 B5 68 | STA | SAVADR | SAVE ADDRESS TO KNOW WHERE TO PUT DATA |
| 1877 A5 65 | LDA | ADRESS+1 | |
| 1879 B5 69 | STA | SAVADR+1 | |
| 187B A5 63 | LDA | LOGCOL | |
| 187D 4B | PHA | | |
| 187E 20 04 F9 | JSR | INCRB8 | PUT CURSOR OVER NEXT CHARACTER |
| 1881 68 | PLA | | |
| 1882 C5 63 | CMP | LOGCOL | TEST NEW LOGCOL AGAINST OLD LOGCOL |
| 1884 B0 10 | BCB | DELCH2 | IF OLD GE NEW THEN QUIT |
| 1886 A5 54 | LDA | ROWCR8 | IS ROW OFF SCREEN? |
| 1888 C0 BF 02 | CMP | ROWCR8 | |
| 188B B0 09 | BCB | DELCH2 | YES, SO QUIT |
| 188D 20 A2 F5 | JBR | GETPLT | GET DATA UNDER CURSOR |
| 1890 A0 00 | LDY | #0 | |
| 1892 91 68 | STA | (SAVADR),Y | PUT IT IN PREVIOUS POSITION |
| 1894 F0 DA | BEQ | DELCH1 | AND LOOP (UNCONDITIONAL) |
| 1896 A0 00 | LDY | #0 | |
| 1898 9B | TYA | | |
| 1899 91 68 | STA | (SAVADR),Y | CLEAR THE LAST POSITION |
| 189B 20 68 FC | JSR | DELTIA | TRY TO DELETE A LINE |
| 189E 20 A8 FC | JSR | PLACR8 | |
| 18A1 4C DD FB | JMP | DOLCOL | AND RETURN |
| 18A4 3B | INSLIN | SEC | NORMAL INSLIN PUTS "1" INTO BIT MAP |
| 18A5 20 78 FB | INSLIA | JBR | EXTEND ENTRY POINT FOR C=0 |
| 18AB A5 52 | LDA | LMARON | DO CARRIAGE RETURN (NO LF) |
| 18AA B5 55 | STA | COLCR8 | |
| 18AC 20 47 F9 | JSR | CONVRT | GET ADDRESS |
| 18AF A5 64 | LDA | ADRESS | SET UP TO=40+FROM (FROM = CURSOR) |
| 18B1 B5 68 | STA | FRMADR | |
| 18B3 1B | CLG | | |
| 18B4 69 2B | ADC | #40 | |
| 18B6 B5 66 | STA | TOADR | |
| 18B8 A5 65 | LDA | ADRESS+1 | |
| 18BA B5 69 | STA | FRMADR+1 | |
| 18BC 69 00 | ADC | #0 | |
| 18BE B5 67 | STA | TOADR+1 | |
| 18C0 A6 54 | LDX | ROWCR8 | SET UP LOOP COUNTER |
| 18C2 E0 17 | CPX | #23 | |
| 18C4 F0 0B | BEQ | INSLI2 | |
| 18C6 20 4E FB | INSLI1 | JSR | MOVLIN |
| 18C9 EB | INX | | |
| 18CA E0 17 | CPX | #23 | |
| 18CC D0 FB | BNE | INSLI1 | |
| 18CE 20 9B FB | INSLI2 | JBR | CLRLIN CLEAR CURRENT LINE |
| 18D1 4C DD FB | JMP | DOLCOL | COLVERT ROW AND COL TO LOGCOL AND RETURN |
| 18D4 20 DD FB | JBR | DOLCOL | GET BEGINNING OF LOG LINE (HOLD1) |
| 18D7 A4 51 | DELLIA | LDY | HOLD1 SQUEEZE BIT MAP |
| 18D9 B4 54 | STY | ROWCR8 | PUT CURSOR THERE |
| 18DB A4 54 | DELLIB | LDY | ROWCR8 |
| 18DD 9B | DELLI1 | TYA | |
| 18DE 3B | SEC | | |
| 18DF 20 23 FB | JSR | LO2GET | GET NEXT BIT |
| 18E2 0B | PHP | | |
| 18E3 9B | TYA | | |
| 18E4 1B | CLC | | |
| 18E6 69 7B | ADC | #120 | |
| 18E7 2B | PLP | | |
| 18E9 20 04 FB | JSR | BITPUT | WRITE IT OVER PRESENT BIT |
| 18EB CB | INY | | |

DISPLAY HANDLER -- 10-30-78 -- DISPLC

| | | | | |
|----------------|--------|-----|----------|--|
| I-BEC C0 18 | | CPY | #24 | |
| I-BEE D0 ED | | BNE | DELL11 | LOOP |
| I-BF0 AD B4 02 | | LDA | LOGMAP+2 | SET LSB |
| I-BF3 09 01 | | DRA | #1 | |
| I-BF5 BD B4 02 | | STA | LOGMAP+2 | |
| I-BF8 A5 52 | DELL12 | LDA | LMARON | DELETE LINE OF DATA USING PART OF SCROLL |
| I-BFA B5 55 | | STA | COLCR5 | CR NO LF |
| I-BFC 20 47 F9 | | JSR | CONVRT | |
| I-BFF 20 B7 FB | | JSR | SCROL1 | |
| I-902 20 20 FB | | JSR | LOGOET | TEST NEXT LINE FOR CONTINUATION |
| | | | | IS IT A NEW LOG LINE? |
| I-909 90 D4 | | BCC | DELL1B | NO SO DELETE ANOTHER |
| I-907 4C DD FB | | JMP | DOLCOL | YES SO DOLCOL AND RETURN |
| I-90A A0 20 | BELL | LDY | #20 | |
| I-90C 20 DB FC | BELL1 | JSR | CLICK | |
| I-90F 88 | | DEY | | |
| I-910 10 FA | | BPL | BELL1 | |
| I-912 60 | | RTB | | |

ROUTINES

DOUBLE BYTE DECREMENT OF INDIRECT POINTER
INCLUDING DB SUBTRACT AND DB DOUBLE DECREMENT

1913 A9 02 DBDDEC LDA #2
1915 D0 0A BNE DBSUB (UNCONDITIONAL)

STORE DATA INDIRECT AND DECREMENT POINTER
(PLACED HERE TO SAVE JMP DBDEC AFTER STORE)

1917 A4 4C STORE LDY DBSTAT RETURN ON ERROR
1919 30 2B BMI STROK
191B A0 00 LDY #0
191D 91 64 STORE1 STA (ADDRESS),Y
JMP DBDEC DECREMENT AND RETURN

191F A9 01 DBDEC LDA #1
1921 BD 9E 02 DBSUB STA SUBTMP
1924 A5 4C LDA DBSTAT RETURN ON ERROR
1926 30 1E BMI STROK

1928 A5 64 LDA ADRESS
192A 38 SEC
192B ED 9E 02 BRC SUBTMP
192E B5 64 STA ADRESS
1930 B0 02 BCS DBSUB1
1932 C6 65 DEC ADRESS+1

1934 A5 0F DBSUB1 LDA APPHI+1 MAKE SURE NOTHING EVER OVERWRITES APPHI
1936 C5 65 CMP ADRESS+1
1938 90 0C BCC STROK OK
193A D0 06 BNE STRERR ERROR
193C A5 0E LDA APPHI
193E C5 64 CMP ADRESS

1940 90 04 BCC STROK
1942 A9 93 STRERR LDA #SCREEN SHOW MEN TOO SMALL FOR SCREEN ERROR
1944 B5 4C STA DBSTAT
1946 60 STROK RTS

CONVERT ROW/COLUMN CURSOR INTO REAL ADDRESS (FROM SAVMSC ON UP)

1947 A5 54 CONVRT LDA ROWCRS SAVE CURSOR
1949 4B PHA
194A A5 55 LDA COLCRS

194C 4B PHA
194D A5 56 LDA COLCRS+1
194F 4B PHA

1950 20 F3 FC JSR PUTMSC
1953 A5 54 LDA ROWCRS PUT 10*ROWCRS INTO MLTTP
1955 B5 66 STA MLTTP

1957 A9 00 LDA #0
1959 B5 67 STA MLTTP+1
195B A5 66 LDA MLTTP QUICK KB

195D 0A ABL A
195E 26 67 ROL MLTTP+1
1960 B5 31 STA HOLD1 (SAVE 2X VALUE)
1962 A4 67 LDY MLTTP+1

| | | | |
|----------------|-----|--|-----|
| 1-764 8C 9F 02 | STY | HOLD2 | " " |
| 1-767 0A | ASL | A | |
| 1-768 26 67 | ROL | MLTTMP+1 | |
| 1-76A 0A | ASL | A | |
| 1-76B 26 67 | ROL | MLTTMP+1 | |
| 1-76D 18 | CLC | ADD IN 2X | |
| 1-76E 65 51 | ADC | HOLD1 | |
| 1-770 85 66 | STA | MLTTMP | |
| 1-772 A5 67 | LDA | MLTTMP+1 | |
| 1-774 6D 9F 02 | ADC | HOLD2 | |
| 1-777 85 67 | STA | MLTTMP+1 | |
| 1-779 A6 87 | LDX | DINDEX NOW SHIFT MLTTMP LEFT DINDEX TIMES TO FINISH | |
| 1-77B 8C 81 FE | LDY | DINDEX, X MULTIPLY | |
| 1-77E 88 | DEY | LOOP N TIMES | |
| 1-77F 30 07 | BMI | CONVR2 | |
| 1-781 06 66 | ASL | MLTTMP | |
| 1-783 26 67 | ROL | MLTTMP+1 | |
| 1-785 4C 7E F9 | JMP | CONVR1 | |
| 1-78B 8C A5 FE | LDY | DIV2TB, X NOW DIVIDE HCRBR TO ACCOUNT FOR PARTIAL BYTES | |
| 1-78B A5 55 | LDA | COLCRB | |
| 1-78D A2 07 | LDX | 07 * TRICKY * | |
| 1-78F 88 | DEY | CONVR3 | |
| 1-790 30 0A | BMI | CONVR4 | |
| 1-792 CA | DEX | | |
| 1-793 46 86 | LBR | COLCRB+1 | |
| 1-795 6A | ROR | A | |
| 1-796 6E A1 02 | ROR | TMPLBT SAVE LOW BITS FOR MASK | |
| 1-799 4C 8F F9 | JMP | CONVR3 | |
| 1-79C C8 | INX | SO Y IS ZERO UPON RETURN FROM THIS ROUTINE | |
| 1-79D 18 | CLC | | |
| 1-79E 65 66 | ADC | MLTTMP ADD SHIFTED COLCRB TO MLTTMP | |
| 1-7A0 85 66 | STA | MLTTMP | |
| 1-7A2 90 02 | BCC | CONVR5 | |
| 1-7A4 E6 67 | INC | MLTTMP+1 | |
| 1-7A6 3B | SEC | * TRICKY * | |
| 1-7A7 6E A1 02 | ROR | TMPLBT SLIDE A BIT UP AGAINST LOW BITS (CONTINUE TILL X=1) | |
| 1-7AA 18 | CLC | | |
| 1-7AB CA | DEX | AND FINISH SHIFT SO LOW BITS ARE | |
| 1-7AC 10 F9 | BPL | CONVR6 RIGHT JUSTIFIED. | |
| 1-7AE AE A1 02 | LDX | TMPLBT IS NOW THE INDEX INTO DMASKTB | |
| 1-7B1 A5 66 | LDA | MLTTMP PREPARE FOR RETURN | |
| 1-7B3 18 | CLC | | |
| 1-7B4 65 66 | ADC | ADDRESS | |
| 1-7B6 85 66 | STA | ADDRESS | |
| 1-7B8 85 5E | STA | OLDADR REMEMBER THIS ADDRESS FOR CURSOR | |
| 1-7BA A5 67 | LDA | MLTTMP+1 | |
| 1-7BC 65 65 | ADC | ADDRESS+1 | |
| 1-7BE 85 65 | STA | ADDRESS+1 | |
| 1-7C0 85 8F | STA | OLDADR+1 | |
| 1-7C2 8D 81 FE | LDA | DMASKT, X | |
| 1-7C5 8D A0 02 | STA | DMASK | |
| 1-7C8 85 6F | STA | SHFAMT | |
| 1-7CA 6B | PLA | | |
| 1-7CB 85 86 | STA | COLCRB+1 | |
| 1-7CD 6B | PLA | | |
| 1-7CE 85 85 | STA | COLCRB | |
| 1-7D0 6B | PLA | | |
| 1-7D1 85 54 | STA | ROWCRB | |
| 1-7D3 60 | RTB | | |

INCREMENT CURSOR AND DETECT BOTH END OF LINE AND END OF SCREEN

| | | | | |
|---------------|---------|-----|-----------|---|
| 19D4 A9 00 | INCRB0 | LDA | #0 | NON-EXTEND ENTRY POINT |
| 19D6 F0 02 | | BEG | INCRSC | |
| 19DB A9 9B | INCRB8 | LDA | #9B | SPECIAL CASE ELIMINATOR |
| 19DA B5 7D | INCRB8 | STA | INSDAT | |
| 19DC E4 43 | INCRSA | INC | LOOCDI | (INSCHE ENTRY POINT) |
| 19DE E4 55 | | INC | COLCR8 | |
| 19E0 D0 02 | | RNE | INCRB2 | DO HIGH BYTE |
| 19E2 E4 54 | | INC | COLCR8+1 | |
| 19E4 A5 55 | INCRB2 | LDA | COLCR8 | TEST END OF LINE |
| 19E6 A6 57 | | LDX | DINDEX | |
| 19E8 D0 8D FE | | CMF | COLUMN, X | TEST TABLED VALUE FOR ALL SCREEN MODES |
| 19EB F0 08 | | BEG | INC2A | DO CR IF EQUAL |
| 19ED E0 00 | | CPX | #0 | MODE 0? |
| 19EF D0 04 | | UNE | INCRB3 | IF NOT, JUST RETURN |
| 19F1 C5 53 | | CMF | RMARON | TEST AGAINST RMARON |
| 19F3 F0 02 | | BEG | INCRB3 | EQUAL IS OK |
| 19F5 B0 01 | | BCB | INC2A | IF GREATER THAN, DO CR |
| 19F7 60 | INCRB3 | RTB | | |
| 19FB E0 08 | INC2A | CPX | #8 | CHECK MODE |
| 19FA 90 04 | | BCC | DOCR1 | NOT 320X1 80 DO IT |
| 19FC A5 54 | | LDA | COLCR8+1 | TEST HSD |
| 19FE F0 F7 | | BEG | INCRB3 | ONLY AT 64 80 DON'T DO IT |
| FA00 A5 57 | DOCR1 | LDA | DINDEX | DONT MESS WITH LOOHP IF NO MODE ZERO |
| FA02 D0 30 | | BNE | DOCR | |
| FA04 A5 63 | | LDA | LOOCDL | TEST LINE OVERRUN |
| FA06 C9 31 | | CMF | #B1 | |
| FA0B 90 0A | | BCC | DOCR1B | IF LESS THAN B1 IT IS DEFINITELY NOT LINE 8 |
| FA0A A5 7D | | LDA | INSDAT | |
| FA0C F0 26 | | BEG | DOCR | ONLY DO LDO LINE OVERFLOW IF INSDAT < 0 |
| FA0E 20 30 FA | | JSR | DOCRWS | LOO LINE OVERFLOW IS SPECIAL CASE |
| FA11 4C 77 FA | | JMP | INCRB1 | RETURN |
| FA14 20 34 FA | DOCR1B | JSR | DOCR | GET IT OVER WITH |
| FA17 A5 84 | | LDA | ROWCR8 | |
| FA19 18 | | CLC | | TEST LOGICAL LINE BIT MAP |
| FA1A 49 7B | | ADC | #120 | |
| FA1C 20 25 FB | | JSR | BITOET | |
| FA1F 90 08 | | BCC | DOCR1A | DONT EXTEND IF OVERRUN IS INTO MIDDLE OF LOO LINE |
| FA21 A5 7D | | LDA | INSDAT | DONT EXTEND IF INSDAT IS ZERO |
| FA23 F0 04 | | BEG | DOCR1A | (INSCHE SPECIAL CASE) |
| FA25 18 | | CLC | | INSERT "0" INTO BIT MAP |
| FA26 20 A5 FB | | JSR | INELIA | |
| FA29 4C DD FB | DOCR1A | JMP | DOLCOL | CONVERT ROW AND COL TO LOOCDL AND RETURN |
| FA2C A9 00 | NOBSCR | LDA | #0 | DOCR WITHOUT SCROLL |
| FA2E F0 02 | | BEG | NOBSCR1 | (UNCONDITIONAL) |
| FA30 A9 9B | DOCRWS | LDA | #9B | DOCR WITH SCROLLING (NORMAL MODE) |
| FA32 B5 7D | NOBSCR1 | STA | INSDAT | |
| FA34 20 E4 FC | DOCR | JSR | COLCR | PLACE COLCR8 AT LEFT EDGE |
| FA37 A9 00 | | LDA | #0 | |
| FA39 B5 56 | | STA | COLCR8+1 | |
| FA3B E4 54 | | INC | ROWCR8 | |
| FA3D A6 57 | DOCR2 | LDX | DINDEX | |
| FA3F A0 18 | | LDY | #24 | SET UP SCROLL LOOP COUNTER |
| FA41 24 7B | | BIT | SHFLO | |
| FA43 10 05 | | BPL | DOCR2A | BRANCH IF NORMAL |
| FA45 A0 04 | | LDY | #4 | |
| FA47 9B | | TYA | | |
| FA4B D0 03 | | BNE | DOCR2B | (UNCONDITIONAL) |

```

1A4A BD 99 FE      DOCR2A LDA    NOROWS, X GET NO OF ROWS
1A4D C5 54          DOCR2B CMP    ROWCRS
1A4F D0 26          BNE     INCRB1
1A51 BC 9D 02          STY     HOLD3
1A54 BA              TXA              DONT SCROLL IF MODE < 0
1A55 D0 20          BNE     INCRB1
1A57 A5 7D          LDA     INSDAT OR IF INSDAT = 0
1A59 F0 1C          BEQ     INCRB1
                      LDA     INSDAT IF INSDAT < $9B THEN ROLL IN A 0
                      CMP     #$9B TO EXTEND BOTTOM LOGICAL LINE
                      BEC
1A5B C9 9B          BEQ     DOCR4B
1A5D 3B              CLC
1A5E F0 01          DOCR4B JSR     SCROLL LOOP BACK TO HERE IF > 1 SCROLLS
1A60 1B              INC     SCRFL0
1A61 20 AC F8      DOCR4B JSR     BUFSTR ROWS MOVE UP SO BUFSTR SHOULD TOO
1A64 EE BB 02          DEC     HOLD3
1A67 C6 AC          DEC     LOGMAP
1A69 CE 9D 02          LDA     LOGMAP FOR PARTIAL LINES, ROLL IN A "1"
1A6C AD B2 02          SEC          DOCR4B AGAIN IF PARTIAL LOGICAL LINE
1A6F 3B              BPL     DOCR4B
1A70 10 EF          LDA     HOLD3 PLACE CURSOR AT NEW LINE NEAR THE BOTTOM
1A72 AD 9D 02          STA     ROWCRS
1A75 B5 54          INCRB1 JMP     DOLCOL COLVERT ROW AND COL TO LOGCOL AND RETURN
1A77 4C DD F8

```

SUBEND: SUBTRACT ENDP1 FROM ROWAC OR COLAC. (X=0 OR 2)

```

1A7A 3B      SUBEND SEC
1A7B B5 70    LDA    ROWAC, X
FA7D E5 74    SBC    ENDP1
FA7F 95 70    STA    ROWAC, X
FAB1 B5 71    LDA    ROWAC+1, X
1AB3 E5 75    SBC    ENDP1+1
1AB5 95 71    STA    ROWAC+1, X
1AB7 60      RTS

```

RANGE: DO CURSOR RANGE TEST. IF ERROR, POP STACK TWICE AND JMP RETURN
(RANGE IS EDITOR ENTRY POINT AND TEST IF EDITOR IS OPEN.
IF IT ISNT IT OPENS THE EDITOR AND CONTINUES)

```

FAB8 AD BF 02      RANGE LDA    BOTSCR IN BOTSCR=4
FAB8 C9 04          CMP     #4
FABD F0 07          BEQ     RANGE THEN IT IS IN MIXED MODE AND OK
1ABF A5 57          LDA     DINDEX IF MODE = 0
1A91 F0 03          BEQ     RANGE THEN IT IS IN EDITOR MODE AND OK
1A93 20 FC F3      JSR     EOPEN IF NOT, OPEN EDITOR
FAB6 A9 27      RANGE LDA     #39 ***** RANGE CHECK RMARON ***** SET UP A0
FAB8 C5 53      CMP     RMARON ***** RANGE CHECK RMARON ***** COMPARE
FAB9 B0 02      BCB     RANGE3 ***** RANGE CHECK RMARON ***** BRANCH OE
1A9C B5 53      STA     RMARON ***** RANGE CHECK RMARON ***** BAD SO STORE 39
1A9E A6 57      RANGE3 LDX     DINDEX
1AA0 BD 99 FE      LDA     NOROWS, X CHECK ROWS
FAA3 C5 54      CMP     ROWCRS
FAA5 90 2A      BCC     RNOERR (ERROR IF TABLE OE, ROWCRS)
FAA7 F0 28      BEQ     RNOERR
1AA9 F0 08      CPX     #0 CHECK FOR 320X1
1AAB D0 0A      BNE     RANGE1 SPECIAL CASE IT
1AAD A5 56      LDA     COLCRS+1
FAAF F0 13      BEQ     RNOOK IF HIGH BYTE IS 0, COL IS OK

```

```

IAB1 C9 01      CMP      #1
IAB3 D0 1C      BNE      RNGERR IF >1, BAD
IAB5 F0 04      BEQ      RNGERR IF 1, GO CHECK LOW BYTE
IAB7 A5 56      RANGE1 LDA      COLC1 FOR OTHERS, NON-ZERO HIGH BYTE IS BAD
IAB9 D0 16      BNE      RNGERR
IABB BD BD FE    RANGE2 LDA      COLUMN, X CHECK LOW BYTE
IABE C5 55      CMP      COLCRB
IAC0 90 0F      BCC      RNGERR
IAC2 F0 0D      BEQ      RNGERR
IAC4 A9 01      RNOOK  LDA      #SUCCEB SET STATUS OK
IAC6 B5 4C      STA      DBSTAT
IAC8 A9 80      LDA      #BRKABT PREPARE BREAK ABORT STATUS
IACA A6 11      LDX      BRKKEY CHECK BREAK KEY FLAG
IACC B5 11      STA      BRKKEY 'CLEAR' BREAK
IACE F0 06      BEQ      RNGERR2 IF BREAK, QUIT IMMEDIATELY AND RETURN TO CIO
IAD0 60      RTS
IAD1 20 D6 F7    RNOERR JSR      HOME ON RANGE ERROR, BRING CURSOR BACK
IAD4 A7 8D      LDA      #CRBROR SHOW CURSOR OVERRANGE ERROR
IAD6 B5 4C      RNOERR2 STA      DBSTAT
IAD8 68      RNOERR1 PLA      RESTORE STACK (THIS ROUTINE IS ALWAYS 1 LEVEL
IAD9 68      PLA      AWAY FROM RETURN TO CIO)
IADA A5 78      LDA      SWPFL0 IF SWAPPED, SWAP BACK
IADC 10 03      BPL      RETUR3
IADE 20 B9 FC    JSR      SWAPA AND DONT DO RETUR1
IAE1 4C 34 F6    RETUR3 JMP      RETUR1 RETURN TO CIO

```

OFFCRS: RESTORE OLD DATA UNDER CURSOR SO IT CAN BE MOVED

```

IAE4 A0 00      OFFCRB LDY      #0
IAE6 A5 3D      LDA      OLDCHR
IAE8 91 5E      STA      (OLDADR),Y
IAEA 60      RTS

```

BITHAP ROUTINES:

```

BITCON: PUT MASK IN BITMSK AND INDEX IN X
BITPUT: PUT CARRY INTO BITHAP
BITROL: ROL CARRY INTO BOTTOM OF BITHAP (SCROLL)
BITSET: SET PROPER BIT
BITCLR: CLEAR PROPER BIT
BITGET: RETURN CARRY SET IF BIT IS THERE
LOGGET: DO BITGET FOR LOGMAP INSTEAD OF TABMAP

```

```

IAEB 48      BITCON PHA
IAEC 29 07    AND      #7
IAEE AA      TAX      GET MASK
IAEF BD B9 FE LDA      MASKTB, X
IAF2 B5 6E    STA      BITMSK
IAF4 68      PLA      PROCESS INDEX
IAF5 4A      LSR      A
IAF6 4A      LSR      A
IAF7 4A      LSR      A
IAFB AA      TAX
IAF9 60      RTS

```


| | | | |
|----------------|--------|-----|----------|
| I-AFA 2E B4 02 | BITROL | ROL | LOGMAP+2 |
| I-AFD 2E B3 02 | | ROL | LOGMAP+1 |
| I-B00 2E B2 02 | | ROL | LOGMAP |
| I-B03 60 | | RTS | |

| | | | |
|-------------|--------|-----|--|
| I-B04 90 0C | BITPUT | BCC | BITCLR AND RETURN OTHERWISE FALL THROUGH TO BITSET AND RETURN |
|-------------|--------|-----|--|

| | | | |
|----------------|--------|-----|-----------|
| I-B06 20 EB FA | BITSET | JSR | BITCON |
| I-B09 BD A3 02 | | LDA | TABMAP, X |
| I-B0C 05 6E | | ORA | BITMSK |
| I-B0E 9D A3 02 | | BTA | TABMAP, X |
| I-B11 60 | | RTS | |

| | | | |
|----------------|--------|-----|-----------|
| I-B12 20 EB FA | BITCLR | JSR | BITCON |
| I-B15 A3 6E | | LDA | BITMSK |
| I-B17 49 FF | | EOR | #FF |
| I-B19 3D A3 02 | | AND | TABMAP, X |
| I-B1C 9D A3 02 | | BTA | TABMAP, X |
| I-B1F 60 | | RTS | |

| | | | |
|-------------|--------|-----|--------|
| I-B20 A5 54 | LOGGET | LDA | ROWCRB |
| I-B22 18 | LOIGET | CLC | |

| | | | |
|----------------|--------|-----|--------|
| I-B23 69 78 | LOGGET | ADC | #120 |
| I-B25 20 EB FA | BITGET | JSR | BITCON |
| I-B28 18 | | CLC | |

| | | | |
|----------------|--|-----|-----------|
| I-B29 BD A3 02 | | LDA | TABMAP, X |
| I-B2C 25 6E | | AND | BITMSK |
| I-B2E F0 01 | | BEG | BITGET |

| | | | |
|----------|--------|-----|--|
| I-B30 38 | | BEC | |
| I-B31 60 | BITGET | RTS | |

INATAC: INTERNAL(CHAR) TO ATAC(ATTACH) CONVERSION

| | | | |
|----------------|--------|-----|------------------------------|
| I-B32 AD FA 02 | INATAC | LDA | CHAR |
| I-B35 A4 57 | | LDV | DINDEX IF GRAPHICS MODES |
| I-B37 C0 03 | | CPY | #3 |
| I-B39 B0 0F | | BCS | INATA1 THEN DONT CHANGE CHAR |

| | | | |
|-------------|--|-----|----|
| I-B3B 2A | | ROL | A |
| I-B3C 2A | | ROL | A |
| I-B3D 2A | | ROL | A |
| I-B3E 2A | | ROL | A |
| I-B3F 29 03 | | AND | #3 |
| I-B41 AA | | TAX | |

| | | | |
|----------------|--------|-----|----------|
| I-B42 AD FA 02 | | LDA | CHAR |
| I-B45 29 9F | | AND | #9F |
| I-B47 1D FA FE | | ORA | INATA, X |
| I-B4A BD FB 02 | INATA1 | STA | ATACHR |
| I-B4D 60 | | RTS | |

MOVLIN: MOVE 40 BYTES AT FRMADR TO TOADR SAVING OLD TOADR
DATA IN THE LINBUF. THEN MAKE NEXT FRMADR
BE AT LINBUF FOR NEXT TRANSFER & TOADR=TOADR+40

| | | | | |
|-------------|--------|-----|-------------|----------------------------|
| I-B4E A9 02 | MOVLIN | LDA | #LINBUF/256 | SET UP ADDRESS=LINBUF-#247 |
|-------------|--------|-----|-------------|----------------------------|


```

I-B50 B5 65      STA      ADDRESS+1
I-B52 A9 47      LDA      #LINDUF
I-B54 B5 64      STA      ADDRESS
I-B56 A0 27      LDY      #39
I-B58 B1 65      MOVL11   LDA      (TOADR),Y      SAVE TO DATA
I-B5A B5 50      STA      TMPCHR
I-B5C B1 68      LDA      (FRMADR),Y      STORE DATA
I-B5E 91 66      STA      (TOADR),Y
I-B60 A5 50      LDA      TMPCHR
I-B62 91 64      STA      (ADDRESS),Y
I-B64 B8         DEY
I-B66 10 F1      BPL      MOVL11
I-B68 A5 65      LDA      ADDRESS+1      SET UP FRMADR=LAST LINE
I-B6A B5 67      STA      FRMADR+1
I-B6C A5 64      LDA      ADDRESS
I-B6E B5 68      STA      FRMADR
I-B70 1B         CLC      ADDD 40 TO TOADR
I-B72 A5 66      LDA      TOADR
I-B74 69 28      ABC      #40
I-B76 B5 66      STA      TOADR
I-B78 90 02      BCC      MOVL12
I-B7A E6 67      INC      TOADR+1
I-B7C 60         MOVL12   RTS

```

EXTEND: EXTEND BIT MAP FROM ROWCRB (EXTEND LOGICAL LINE)

```

I-B7B 0B         EXTEND   PHP      SAVE CARRY
I-B7C A0 17      LDY      #23
I-B7E 98         EXTEN1   TYA
I-B7F 20 22 FB   JBR      LOI0ET
I-B82 0B         PHP
I-B83 98         TYA
I-B84 1B         CLC
I-B86 69 79      ADC      #121
I-B88 28         PLP
I-B8A 20 04 FB   JBR      BITPUT
I-B8C B8         EXTEN3   DEY
I-B8E 30 04      BMI      EXTEN4
I-B90 C4 54      CPY      ROWCRB
I-B92 B0 EC      BCB      EXTEN1
I-B94 A5 54      EXTEN4   LDA      ROWCRB
I-B96 1B         CLC
I-B98 69 78      ADC      #120
I-B9A 28         PLP
I-B9C 4C 04 FB   JMP      BITPUT  STORE NEW LINE'S BIT AND RETURN

```

CLRLIN: CLEAR LINE CURSOR IS ON

```

I-B9B A5 52      CLRLIN   LDA      LMARON
I-B9D B5 55      STA      COLCRB
I-B9F 20 47 F9   JBR      CONVRT
I-BA2 A0 27      LDY      #39
I-BA4 A9 00      LDA      #0
I-BA6 91 64      CLRL11   STA      (ADDRESS),Y
I-BA8 B8         DEY
I-BA9 10 FB      DPL      CLRL11

```

FBAB 60

RTS

SCROLL: SCROLL SCREEN

| | | | | |
|---------------|--------|-----|--------------|---|
| FBAC 20 FA FA | SCROLL | JBR | BITROL | ROLL IN CARRY |
| FBAF A5 58 | | LDA | SAVMSB | SET UP WORKING REGISTERS |
| FBBI B5 64 | | STA | ADDRESS | |
| FBH3 A5 59 | | LDA | SAVMSB+1 | |
| FBH5 B5 65 | | STA | ADDRESS+1 | |
| FBH7 A0 28 | SCROL1 | LDY | #40 | LOOP |
| FBH9 B1 64 | | LDA | (ADDRESS), Y | |
| FBHB A6 6A | | LDX | RANTOP | TEST FOR LAST LINE |
| FBHD CA | | DEX | | |
| FBHE E4 63 | | CPX | ADDRESS+1 | |
| FBCH D0 08 | | BNE | SCROL2 | |
| FBCE A2 D7 | | LDX | #D7 | |
| FBCH E4 64 | | CPX | ADDRESS | |
| FBCH B0 02 | | BCB | SCROL2 | |
| FBCH A9 00 | | LDA | #0 | YES SO STORE ZERO DATA FOR THIS ENTIRE LINE |
| FBCH A0 00 | SCROL2 | LDY | #0 | |
| FBCH 91 64 | | STA | (ADDRESS), Y | |
| FBCE E6 64 | | INC | ADDRESS | |
| FBCH D0 E5 | | BNE | SCROL1 | |
| FBCH E6 65 | | INC | ADDRESS+1 | |
| FBCH A5 65 | | LDA | ADDRESS+1 | |
| FBCH C5 6A | | CHP | RANTOP | |
| FBCH D0 DD | | BNE | SCROL1 | |
| FBCH AC DD FB | | JMP | DOLCOL | AND RETURN |

DOLCOL: DO LOGICAL COLUMN FROM BITMAP AND COLCRB

| | | | | |
|---------------|---------|-----|---------|--------------------------|
| FBCH A9 00 | DOLCOL | LDA | #0 | START WITH ZERO |
| FBCH 85 63 | | STA | LOGCOL | |
| FBCH A5 54 | | LDA | ROWCRB | |
| FBCH 85 51 | | STA | HOLD1 | |
| FBCH A5 51 | DOLCOL1 | LDA | HOLD1 | ADD IN ROW COMPONENT |
| FBCH 20 22 FB | | JBR | LOISET | |
| FBCH B0 00 | | BCB | DOLCOL2 | FOUND BEGINNING OF LINE |
| FBCH A5 63 | | LDA | LOGCOL | ADD 40 AND LOOK BACK ONE |
| FBCH 18 | | CLC | | |
| FBCH 69 28 | | ADC | #40 | |
| FBCH 85 63 | | STA | LOGCOL | |
| FBCH C6 51 | | DEC | HOLD1 | UP ONE LINE |
| FBCH 4C E5 FB | | JMP | DOLCOL1 | |
| FBCH 18 | DOLCOL2 | CLC | | ADD IN COLCRB |
| FBCH A5 63 | | LDA | LOGCOL | |
| FBCH 65 55 | | ADC | COLCRB | |
| FBCH 85 63 | | STA | LOGCOL | |
| FBCH 60 | | RTS | | |

DOBUFC: COMPUTE BUFFER COUNT AS THE NUMBER OF BYTES FROM
BUFBTR TO END OF LOGICAL LINE WITH TRAILING SPACES REMOVED

FC00 20 9D FC DOBUFC JSR PHACRB

```

FC03 A5 63      LDA      LOGCOL
FC03 48          PHA
FC06 A5 6C      LDA      BUFSTR START
FC08 B5 54      STA      ROWCRS
FC0A A5 6D      LDA      BUFSTR+1
FC0C B5 55      STA      COLCRS
FC0E A9 01      LDA      #1
FC10 B5 68      STA      BUFCNT
FC12 A2 17      DOBUF1  LDX      #23      NORMAL
FC14 A5 78      LDA      SWPFL0 IF SWAPPED, ROW 3 IS THE LAST LINE ON SCREEN
FC16 10 02      BPL      DOB1
FC18 A2 03      LDX      #3
FC1A E4 54      DOB1     CPX      ROWCRS TEST IF CRSR IS AT LAST SCREEN POSITION
FC1C D0 08      BNE      DOBU1A
FC1E A5 55      LDA      COLCRS
FC20 C5 53      CMP      RMARON
FC22 D0 05      BNE      DOBU1A
FC24 E6 6B      INC      BUFCNT YES SO FARE INCRSR TO AVOID SCROLLING
FC26 4C 39 FC    JMP      DOBUF2
FC28 20 D4 F9    DOBU1A  JSR      INCRSR
FC2C E6 6B      INC      BUFCNT
FC2E A5 63      LDA      LOGCOL
FC30 C5 52      CMP      LMARON
FC32 D0 DE      BNE      DOBUF1 NOT YET EOL
FC34 C6 54      DEC      ROWCRS BACK UP ONE INCRSR
FC36 20 99 F7    JSR      CRRLF
FC38 20 A2 F5    DOBUF2  JSR      GETPLT TEST CURRENT COLUMN FOR NON-ZERO DATA
FC3C D0 17      BNE      DOBUF4 QUIT IF NON-ZERO
FC3E C6 68      DEC      BUFCNT DECREMENT COUNTER
FC40 A5 63      LDA      LOGCOL BEGINNING OF LOGICAL LINE YET?
FC42 C5 52      CMP      LMARON
FC44 F0 0F      BEQ      DOBUF4 YES, SO QUIT
FC46 20 99 F7    JSR      CRRLF BACK UP CURSOR
FC48 A5 55      LDA      COLCRS IF LOGCOL=RMARON, GO UP 1 ROW
FC4B C5 53      CMP      RMARON
FC4D D0 02      BNE      DOBUF3
FC4F C6 54      DEC      ROWCRS
FC51 A5 68      DOBUF3  LDA      BUFCNT
FC53 D0 E4      BNE      DOBUF2 LOOP UNLESS BUFCNT JUST WENT TO ZERO
FC55 68          DOBUF4  PLA
FC56 B5 63      STA      LOGCOL
FC58 20 AB FC    JSR      PLACRS
FC5B 60          RTS

```

STRREQ: MOVE BUFSTR TO BEGINNING OF LOGICAL LINE

```

FC5D 20 DD FB    STRREQ  JSR      DOLCOL USE DOLCOL TO POINT HOLD1 AT EOL
FC5F A5 51      LDA      HOLD1
FC61 B5 6C      STA      BUFSTR
FC63 A5 52      LDA      LMARON
FC65 B5 6D      STA      BUFSTR+1
FC67 60          RTS

```

DELTIM: TIME TO DELETE A LINE IF IT IS EMPTY AND AN EXTENSION

| | | | | |
|---------------|--------|-----|------------|----------------------------------|
| IC6B A5 63 | DELTIA | LDA | LOGCOL | IF LOGCOL < LMARON |
| IC6A C5 52 | | CMP | LMARON | THEN DONT MOVE UP ONE |
| IC6C D0 02 | | BNE | DELTID | LINE BEFORE TESTING DELTIM |
| IC6E C6 54 | | DEC | ROWCRB | |
| IC70 20 D0 FB | DELTIB | JSR | DOLCOL | |
| IC73 A5 63 | DELTIM | LDA | LOGCOL | TEST FOR EXTENSION |
| IC75 C5 52 | | CMP | LMARON | |
| IC77 F0 13 | | BEQ | DELTIC | NO |
| IC79 20 47 F9 | | JSR | CONVRT | |
| IC7C A5 53 | | LDA | RMARON | SET UP COUNT |
| IC7E 38 | | BEC | | |
| IC7F E3 52 | | BBC | LMARON | |
| IC81 A8 | | TAY | | |
| IC82 D1 64 | DELTII | LDA | (ADRE88),Y | |
| IC84 D0 06 | | DNE | DELTIC | FOUND A NON-0 SO QUIT AND RETURN |
| IC86 BB | | DEY | | |
| IC87 10 F9 | | BPL | DELTII | |
| IC89 4C DB FB | DELTIC | JMP | DELTIB | DELETE A LINE AND RETURN |
| IC8C 60 | DELTIC | RTS | | |

TSTCTL: SEARCH CNTRLB TABLE TO SEE IF ATACHR IS A CNTRL CHAR

| | | | | |
|---------------|--------|-----|----------|-------------------------|
| IC8D A2 2D | TSTCTL | LDX | #45 | PREPARE TO SEARCH TABLE |
| IC8F BD C6 FE | TSTCT1 | LDA | CNTRLB,X | |
| IC92 CD FB 02 | | CMP | ATACHR | |
| IC95 F0 05 | | BEQ | TSTCT2 | |
| IC97 CA | | DEX | | |
| IC98 CA | | DEX | | |
| IC99 CA | | DEX | | |
| IC9A 10 F3 | | BPL | TSTCT1 | |
| IC9C 60 | TSTCT2 | RTB | | |

PUSH ROWCRB, COLCRB AND COLCRB+1

| | | | |
|---------------|--------|-----|----------|
| IC9D A2 02 | PHACRB | LDX | #2 |
| IC9F B5 54 | PHACR1 | LDA | ROWCRB,X |
| ICA1 9D BB 02 | | STA | TMPCRB,X |
| ICA4 CA | | DEX | |
| ICA5 10 FB | | BPL | PHACR1 |
| ICA7 60 | | RTB | |

PULL COLCRB+1, COLCRB AND ROWCRB

| | | | |
|---------------|--------|-----|----------|
| ICAB A2 02 | PLACRB | LDX | #2 |
| ICAA BD BB 02 | PLACR1 | LDA | TMPCRB,X |
| ICAD 95 54 | | STA | ROWCRB,X |
| ICAF CA | | DEX | |
| ICBQ 10 FB | | BPL | PLACR1 |
| ICB2 60 | | RTS | |

SWAP: IF MIXED MODE, SWAP TEXT CURSORB WITH REGULAR CURSORB

| | | | | |
|---------------|-------|-----|----------|------------------------------|
| ICB3 20 B9 FC | SWAP | JBR | SWAPA | THIS ENTRY POINT DOES RETURN |
| ICB6 4C 34 F6 | | JMP | RETURN | |
| ICD9 AD BF 02 | SWAPA | LDA | BOTSCR | |
| ICDC C9 1B | | CMP | #24 | |
| ICDE F0 17 | | BEQ | SWAP3 | |
| ICC0 A2 08 | | LDX | #11 | |
| ICC2 B5 54 | SWAP1 | LDA | ROWCRS1 | |
| ICC4 4B | | PHA | | |
| ICC5 BD 90 02 | | LDA | TXTRW.X | |
| ICCB 95 54 | | STA | ROWCRS.X | |
| ICCA 6B | | PLA | | |
| ICCB 9D 90 02 | | STA | TXTRW.X | |
| ICCE CA | | DEX | | |
| ICCF 10 F1 | | RPL | SWAP1 | |
| ICD1 A5 7B | | LDA | SWPFLO | |
| ICD3 49 FF | | EOR | #FF | |
| ICD5 B5 7B | | STA | SWPFLO | |
| ICD7 60 | SWAP3 | RTB | | |

CLICK: MAKE CLICK THROUGH KEYBOARD SPEAKER

| | | | |
|---------------|--------|-----|---------|
| ICDB A2 7F | CLICK | LDX | #7F |
| ICDA 8E 1F D0 | CLICK1 | STX | CONSOLE |
| ICDD 8E 0A D4 | | STX | HSYNC |
| ICE0 CA | | DEX | |
| ICE1 10 F7 | | BPL | CLICK1 |
| ICE3 60 | | RTB | |

COLCR: PUTS EITHER 0 OR LMARON INTO COLCRS BASED ON MODE AND SWPFLO

| | | | |
|------------|--------|-----|--------|
| ICE4 A9 00 | COLCR | LDA | #0 |
| ICE6 A6 7B | | LDX | SWPFLO |
| FCEB D0 04 | | BNE | COLCR1 |
| FCEA A4 57 | | LDX | DINDEX |
| FCEC D0 02 | | BNE | COLCR2 |
| ICEE A5 52 | COLCR1 | LDA | LMARON |
| ICF0 B5 55 | COLCR2 | STA | COLCRS |
| ICF2 60 | | RTB | |

PUTMBC: PUT SAVMBC INTO ADDRESS

| | | | | |
|------------|--------|-----|-----------|----------------|
| ICF3 A5 5B | PUTMBC | LDA | SAVMBC | SET UP ADDRESS |
| ICF5 B5 64 | | STA | ADDRESS | |
| ICF7 A5 59 | | LDA | SAVMBC+1 | |
| ICF9 B5 65 | | STA | ADDRESS+1 | |
| ICF8 60 | | RTB | | |

```

DRAW -- DRAW A LINE FROM OLDROW,OLDCOL TO NEWROW,NEWCOL
( THE AL MILLER METHOD FROM BASKETBALL )

ICFC A2 00      DRAW      LDX      #0
ICFE A5 22      LDA      ICCOMZ  TEST COMMAND: #11-DRAW  #12-FILL
ID00 C9 11      CMP      #11
ID02 F0 08      BEQ      DRAWA
ID04 C9 12      CMP      #12  TEST FILL
ID06 F0 03      BEQ      DRAWB  YES
ID08 A0 84      LDY      INVALID NO, SO RETURN INVALID COMMAND
ID0A 60
ID0B E8          DRAWB      INX
ID0C 8E 87 02    DRAWA     STX      FILFLO
ID0F A5 54      LDA      ROWCR8  PUT CURSOR INTO NEWROW,NEWCOL
ID11 85 60      STA      NEWROW
ID13 A5 55      LDA      COLCR8
ID15 85 61      STA      NEWCOL
ID17 A5 64      LDA      COLCR8+1
ID19 85 62      STA      NEWCOL+1
ID1B A9 01      LDA      #1
ID1D 85 79      STA      ROWINC  SET UP INITIAL DIRECTIONS
ID1F 85 7A      STA      COLINC

FD21 38          SEC
FD22 A5 60      LDA      NEWROW  DETERMINE DELTA ROW
FD24 E9 8A      SBC      OLDROW
ID26 85 76      STA      DELTAR
ID28 80 00      BCS      DRAW1    DO DIRECTION AND ABSOLUTE VALUE
ID2A A9 FF      LDA      #FF      BORROW HAS ATTEMPTED
FD2C 85 79      STA      ROWINC  SET DIRECTION=DOWN
ID2E A5 76      LDA      DELTAR
FD30 A9 FF      EOR      #FF      DELTAR = |DELTAR|
FD32 18          CLC
FD33 69 01      ADC      #1
FD35 85 76      STA      DELTAR

FD37 38          DRAW1     SEC
FD38 A5 61      LDA      NEWCOL  NOW DELTA COLUMN
FD3A E5 58      SBC      OLDCOL
ID3C 85 77      STA      DELTAC
ID3E A5 62      LDA      NEWCOL+1  TWO-BYTE QUANTITY
ID40 E5 5C      SBC      OLDCOL+1
FD42 85 78      STA      DELTAC+1
ID44 80 14      BCS      DRAW2    DIRECTION AND ABSOLUTE VALUE
ID46 A9 FF      LDA      #FF      BORROW HAS ATTEMPTED
ID48 85 7A      STA      COLINC  SET DIRECTION = LEFT
ID4A A5 77      LDA      DELTAC
ID4C A9 FF      EOR      #FF      DELTAC = |DELTAC|
FD4E 85 77      STA      DELTAC
FD50 A5 78      LDA      DELTAC+1
FD52 A9 FF      EOR      #FF
FD54 85 78      STA      DELTAC+1
FD56 E6 77      INC      DELTAC  ADD ONE FOR TWO'S COMPLEMENT
FD58 D0 02      BNE      DRAW2

FD5A E6 78      INC      DELTAC+1
FD5C A2 02      DRAW2     LDX      #2  ZERO RAM FOR DRAW LOOP
FD5E A0 00      LDY      #0
ID60 84 73      STY      COLAC+1
FD62 98          DRAW3A    TYA
FD63 95 70      STA      ROWAC, X
FD65 85 5A      LDA      OLDROW, X

```


| | | | |
|----------------|-----|-----------|--------------------------------------|
| I-D67 98 54 | STA | ROWCRB, X | |
| I-D69 CA | DEY | | |
| I-D6A 10 F6 | LDA | DRAW3A | |
| I-D6C A5 77 | STA | DELTA | IF THE LARGER ONE (ROW OR COL) |
| | STA | COUNTR | COMPARE COUNTR AND ENDP |
| | STA | ENDPT | |
| I-D6E E8 | INX | | |
| I-D6F AB | TAY | | |
| I-D70 A5 78 | LDA | DELTA | |
| I-D72 B5 7F | STA | COUNTR+1 | |
| I-D74 05 75 | STA | ENDPT+1 | |
| I-D76 D0 08 | BNE | DRAW3 | AUTOMATICALLY LARGER IF MBD>0 |
| I-D78 A5 77 | LDA | DELTA | |
| I-D7A C5 76 | CMF | DELTAR | LOW COL > LOW ROW? |
| I-D7C B0 05 | BCC | DRAW3 | YES |
| I-D7E A5 76 | LDA | DELTAR | |
| I-D80 A2 02 | LDX | #2 | |
| I-D82 AB | TAY | | |
| I-D83 98 | TVA | | PUT IN INITIAL CONDITIONS |
| I-D84 B5 7E | STA | COUNTR | |
| I-D86 17 74 | STA | ENDPT | |
| I-D88 4B | PHA | | SAVE AC |
| I-D89 A5 75 | LDA | ENDPT+1 | PUT LBB OF HIGH BYTE |
| I-D8B 4A | LBR | A | INTO CARRY |
| I-D8C 6B | PLA | | RESTORE AC |
| I-D8D 6A | ROB | A | FOR THE 9 BIT ACCUMULATOR |
| I-D8E 95 70 | STA | ROWAC, X | |
| I-D90 A5 7E | LDA | COUNTR | TEST ZERO |
| I-D92 05 7F | ORA | COUNTR+1 | |
| I-D94 D0 03 | BNE | DRAW11 | IF COUNTR IS ZERO, LEAVE DRAW |
| I-D96 4C 42 FE | JMP | DRAW10 | |
| I-D97 1B | CLC | | ADD ROW TO ROWAC (PLOT LOOP) |
| I-D9A A5 70 | LDA | ROWAC | |
| I-D9C 65 76 | ADC | DELTAR | |
| I-D9E B5 70 | STA | ROWAC | |
| I-DA0 90 02 | BCC | DRAW5 | |
| I-DA2 E6 71 | INC | ROWAC | |
| I-DA4 A5 71 | LDA | ROWAC+1 | COMPARE ROW TO ENDP |
| I-DA6 C5 73 | CMF | ENDPT+1 | IF HIGH BYTE OF ROW IS LT. HIGH |
| I-DAB 90 14 | BCC | DRAW6 | BYTE OF ENDP, BLT TO COLUMN |
| I-DA8 D0 06 | BNE | DRAW3A | |
| I-DAC A5 70 | LDA | ROWAC | |
| I-DAE C5 74 | CMF | ENDPT | LOW BYTE |
| I-DB0 90 0C | BCC | DRAW6 | ALSO BLT |
| I-DB2 1B | CLC | | OE SO MOVE POINT |
| I-DB3 A5 54 | LDA | ROWCRB | |
| I-DB5 65 79 | ADC | ROWINC | |
| I-DB7 B5 54 | STA | ROWCRB | |
| I-DB9 A2 00 | LDX | #0 | AND SUBTRACT ENDP FROM ROWAC |
| I-DBB 20 7A FA | JBR | SUBEND | |
| I-DBE 1B | CLC | | DO SAME FOR COLUMN (DOUBLE BYTE ADD) |
| I-DBF A5 72 | LDA | COLAC | ADD |
| I-DC1 65 77 | ADC | DELTAC | |
| I-DC3 B5 72 | STA | COLAC | |
| I-DC5 A5 73 | LDA | COLAC+1 | |
| I-DC7 65 78 | ADC | DELTAC+1 | |
| I-DC9 B5 73 | STA | COLAC+1 | |
| I-DCB C5 75 | CMF | ENDPT+1 | COMPARE HIGH BYTE |
| I-DCD 90 27 | BCC | DRAWB | |
| I-DCF D0 06 | BNE | DRAW6A | |

| | | | | |
|----------------|--------|-----|----------|--------------------------------------|
| I-DD1 A5 72 | | LDA | COLAC | COMPARE LOW BYTE |
| I-DD3 C5 74 | | CHP | ENDPT | |
| I-DD5 90 1F | | RCC | DRAWB | |
| I-DD7 24 7A | DRAW6A | BIT | COLINC | + OR - ? |
| I-DD9 10 10 | | BPL | DRAW6B | |
| I-DDB C6 55 | | DEC | COLCRS | DO DOUBLE BYTE DECREMENT |
| I-DDD A5 55 | | LDA | COLCRS | |
| I-DDF C9 FF | | CHP | #\$FF | |
| I-DE1 D0 0E | | BNE | DRAW7 | |
| I-DE3 A5 56 | | LDA | COLCRS+1 | |
| I-DE5 F0 0A | | BEQ | DRAW7 | DONT DEC IF ZERO |
| I-DE7 C6 56 | | DEC | COLCRS+1 | |
| I-DE9 10 04 | | BPL | DRAW7 | (UNCONDITIONAL) |
| I-DEB E6 55 | DRAW6B | INC | COLCRS | DO DOUBLE BYTE INCREMENT |
| I-DED D0 02 | | BNE | DRAW7 | |
| I-DEF E6 56 | | INC | COLCRS+1 | |
| I-DF1 A2 02 | DRAW7 | LDX | #2 | AND SUBTRACT ENDPT FROM COLAC |
| I-DF3 20 7A FA | | JBR | BUBEND | |
| I-DF6 20 76 FA | DRAWB | JBR | RANGE | |
| I-DF9 20 E0 F5 | | JBR | OUTPLT | PLOT POINT |
| I-DFC AD D7 02 | | LDA | FILFLO | TEST RIGHT FILL |
| I-DFE F0 2F | | BEQ | DRAW9 | |
| I-E01 20 9D FC | | JSR | PHACRS | |
| I-E04 AD FB 02 | | LDA | ATACHR | |
| I-E07 BD BC 02 | | STA | HOLD4 | |
| I-E0A A5 B4 | DRAWBA | LDA | ROWCRS | SAVE ROW IN CASE OF CR |
| I-E0C 48 | | PHA | | |
| I-E0D 20 DC F9 | | JBR | INCRSA | POSITION CURSOR ONE PAST DOT |
| I-E10 68 | | PLA | | RESTORE ROWCRS |
| I-E11 B5 54 | | STA | ROWCRS | |
| I-E13 20 76 FA | DRAWB | JBR | RANGE | |
| I-E16 20 A2 F5 | | JBR | OUTPLT | GET DATA |
| I-E19 D0 0C | | BNE | DRAWBB | STOP IF NON-ZERO DATA IS ENCOUNTERED |
| I-E1B AD FD 02 | | LDA | FILDAT | FILL DATA |
| I-E1E BD FB 02 | | STA | ATACHR | |
| I-E21 20 E0 F5 | | JBR | OUTPLT | DRAW LOOP |
| I-E24 4C 0A FE | | JMP | DRAWBA | |
| I-E27 AD BC 02 | DRAWBB | LDA | HOLD4 | |
| I-E2A BD FB 02 | | STA | ATACHR | |
| I-E2D 20 AB FC | | JSR | PLACRS | |
| I-E30 38 | DRAW9 | BEC | | DO DOUBLE BYTE SUBTRACT |
| I-E31 A5 7E | | LDA | COUNTR | |
| I-E33 E9 01 | | BBC | #1 | |
| I-E35 B5 7E | | STA | COUNTR | |
| I-E37 A5 7F | | LDA | COUNTR+1 | |
| I-E39 E9 00 | | BBC | #0 | |
| I-E3B B5 7F | | STA | COUNTR+1 | |
| I-E3D 00 03 | | BMI | DRAW10 | |
| I-E3F 40 90 FD | | JMP | DRAW4A | |
| I-E42 4C 34 F6 | DRAW10 | JMP | RETURN | |

MEMORY ALLOCATION

NUMBER OF DISPLAY LIST ENTRIES

MXDMDE BYTE 19,19,9,19,39,39,79,79,65,65,65,65 (EXTENSION OF NMDLE)

ANTIC CODE FROM INTERNAL MODE CONVERSION TABLE

| INTERNAL | ANTIC CODE | DESCRIPTION |
|----------|------------|--|
| 0 | 2 | 40X2X8 CHARACTERS |
| 1 | 6 | 20X5X8 " |
| 2 | 7 | 20X5X16 " |
| 3 | 8 | 40X4X8 GRAPHICS |
| 4 | 9 | 80X2X4 " |
| 5 | A | 80X4X4 " |
| 6 | B | 160X2X2 " |
| 7 | D | 160X4X2 " |
| 8 | F | 320X2X1 " |
| 9 | | SAME AS 8 BUT QTIA 'LUM' MODE |
| 10 | | SAME AS 8 BUT QTIA 'COL/LUM REGISTER' MODE |
| 11 | | SAME AS 8 BUT QTIA 'COLOR' MODE |

FE69 02 04 07 ANCONV BYTE 2, 4, 7, 8, 9, 1A, 1B, 1D, 1F, 1F, 1F, 1F ZEROS FOR RANGE TEST IN PAGE 1
FE6C 08 09 0A
FE6F 0B 0D 0F
1E72 0F 0F 0F

PAGE TABLE TELLS WHICH DISPLAY LISTS ARE IN DANGER OF CROSSING A 256 BYTE PAGE BOUNDARY

| | | | |
|---------------|---------|------|-------------------------|
| FE75 00 00 00 | PAGE TB | BYTE | 0,0,0,0,0,0,0,1,1,1,1,1 |
| FE78 00 00 00 | | | |
| FE7B 00 01 01 | | | |
| FE7E 01 01 01 | | | |

THIS IS THE NUMBER OF LEFT SHIFTS NEEDED TO MULTIPLY
COLCR8 BY 10, 20, OR 40. (ROWCR8*10)/(2**DHLINE)

FEB1 02 01 01 DLINE BYTE 2.1.1.0.0.1.1.2.2.2.2.2

IEB4 00 00 01
IEB7 01 02 02
IEB8 02 02 02

COLUMN: NUMBER OF COLUMNS

IEB8 28 14 14
IE90 28 50 50
IE93 A0 A0 40
IE96 50 50 50

COLUMN . BYTE 40, 20, 20, 40, 80, 80, 160, 160, 64, 80, 80, 80 MODE 8 IS SPECIAL CASE

NOROWS: NUMBER OF ROWS

IE99 18 18 0C
IE9C 18 30 30
IE9F 60 60 0C
IEA2 0C 0C 0C

NOROWS . BYTE 24, 24, 12, 24, 48, 48, 96, 96, 192, 192, 192, 192

DIV2TB: HOW MANY RIGHT SHIFTS FOR HCRBR FOR PARTIAL BYTE MODES

IEA5 00 00 00
IEA8 02 03 02
IEA8 03 02 03
IEAE 01 01 01

DIV2TB . BYTE 0, 0, 0, 2, 3, 2, 3, 2, 3, 1, 1, 1

DMASKT: DISPLAY MASK TABLE

IEB1 00 FF F0
IEB4 0F
IEB5 00 30 00
IEB8 03

DMASKT . BYTE \$00, \$FF, \$F0, \$0F

DMASKT . BYTE \$00, \$30, \$00, \$03

MASKTB: BIT MASK. (ALSO PART OF DMASKTB! DO NOT SEPARATE)

IEB9 80 40 20
IEBC 10 08 04
IEBF 02 01

MASKTB . BYTE \$80, \$40, \$20, \$10, \$08, \$04, \$02, \$01

IEC1 28 CA 94
IEC4 46 00

COLRTB . BYTE \$28, \$CA, \$94, \$46, \$00

CNTRLB: CONTROL CODES AND THEIR DISPLACEMENTS INTO THE CONTROL CHARACTER PROCESSORS

IEC6 18
IEC7 79 F7
IEC9 1C

CNTRLB . BYTE \$18
CNTRLB . WORD ESCAPE
CNTRLB . BYTE \$1C

| | | |
|------------|------|--------|
| IECA 7F F7 | WORD | CRGRUP |
| IECC 1D | BYTE | \$1D |
| IECD 8C F7 | WORD | CRSRDN |
| IECF 1E | BYTE | \$1E |
| IED0 99 F7 | WORD | CRSRLF |
| IED2 1F | BYTE | \$1F |
| IED3 AA F7 | WORD | CRSRRT |
| IED5 7D | BYTE | \$7D |
| IED6 89 F7 | WORD | CLRSR |
| IED8 7E | BYTE | \$7E |
| IED9 E6 F7 | WORD | BS |
| IEDB 7F | BYTE | \$7F |
| IEDC 10 F8 | WORD | TAB |
| IEDF 9B | BYTE | \$9B |
| IEDF 30 FA | WORD | DOCRWS |
| IEE1 9C | BYTE | \$9C |
| IEE2 D4 F8 | WORD | DELLIN |
| IEE4 9D | BYTE | \$9D |
| IEE5 A4 F8 | WORD | INSLIN |
| IEE7 9E | BYTE | \$9E |
| IEE8 32 F8 | WORD | CLRTAB |
| IEEA 9F | BYTE | \$9F |
| IEEB 2D F8 | WORD | SETTAB |
| IEED FD | BYTE | \$FD |
| IEEE 0A F9 | WORD | BELL |
| IEF0 FE | BYTE | \$FE |
| IEF1 6D F8 | WORD | DELCHR |
| IEF3 FF | BYTE | \$FF |
| IEF4 37 F8 | WORD | INSCHR |

ATAINT: ATASCII TO INTERNAL TABLE

| | | | |
|---------------|--------|------|------------------------|
| IEF6 40 00 20 | ATAINT | BYTE | \$40, \$00, \$20, \$40 |
| IEF9 60 | | | |

INTATA: INTERNAL TO ATASCII TABLE

| | | | |
|---------------|--------|------|------------------------|
| IEFA 20 40 00 | INTATA | BYTE | \$20, \$40, \$00, \$40 |
| IEFD 60 | | | |

ATASCI: ATASCII CONVERSION TABLE

| | | | |
|---------------|--------|--|---|
| IEFE 6C 6A 08 | ATASCI | BYTE | \$6C, \$6A, \$38, \$80, \$80, \$68, \$2B, \$2A LOWER CASE |
| IF01 80 80 68 | | | |
| IF04 2B 2A | | | |
| IF06 6F 80 70 | BYTE | \$6F, \$80, \$70, \$75, \$78, \$69, \$2D, \$3D | |
| IF09 75 9B 69 | | | |
| FF0C 2D 3D | | | |
| FF0E 76 80 63 | BYTE | \$76, \$80, \$63, \$80, \$80, \$62, \$78, \$7A | |
| IF11 80 80 62 | | | |
| IF14 78 7A | | | |
| IF16 34 80 33 | BYTE | \$34, \$80, \$33, \$36, \$1B, \$35, \$32, \$31 | |
| FF19 36 1B 39 | | | |

IF1C 32 31

IF1E 2C 20 2E . BYTE \$2C, \$20, \$2E, \$6E, \$80, \$4D, \$2F, \$81

IF21 6E 80 6D

IF24 2F 81

IF26 72 80 65 . BYTE \$72, \$80, \$65, \$79, \$7F, \$74, \$77, \$71

IF29 79 7F 74

IF2C 77 71

IF2E 39 80 30 . BYTE \$39, \$80, \$30, \$37, \$7E, \$38, \$3C, \$3E

IF31 37 7E 38

IF34 3C 3E

IF36 66 68 64 . BYTE \$66, \$68, \$64, \$80, \$82, \$67, \$73, \$61

IF39 80 82 67

IF3C 73 61

IF3E 40 4A 3A . BYTE \$4C, \$4A, \$3A, \$80, \$80, \$4B, \$50, \$5E UPPER CASE

IF41 80 80 48

IF44 5C 5E

IF46 4F 80 50 . BYTE \$4F, \$80, \$50, \$55, \$9B, \$49, \$5F, \$7C

IF49 55 9B 49

IF4C 5F 7C

IF4E 56 80 43 . BYTE \$56, \$80, \$43, \$80, \$80, \$42, \$5B, \$5A

IF51 80 80 42

IF54 5B 5A

IF56 24 80 23 . BYTE \$24, \$80, \$23, \$26, \$1B, \$25, \$22, \$21

IF59 26 1B 25

IF5C 22 21

IF5E 5B 20 5D . BYTE \$5B, \$20, \$5D, \$4E, \$80, \$4D, \$3F, \$81

IF61 4E 80 4D

IF64 3F 81

IF66 52 80 45 . BYTE \$52, \$80, \$45, \$59, \$9F, \$54, \$57, \$51

IF69 59 9F 54

IF6C 57 51

IF6E 2B 80 29 . BYTE \$2B, \$80, \$29, \$27, \$9C, \$40, \$7D, \$9D

IF71 27 9C 40

IF74 7D 9D

IF76 46 4B 44 . BYTE \$46, \$4B, \$44, \$80, \$80, \$47, \$43, \$41

IF79 80 83 47

IF7C 53 41

IF7E 0C 0A 7B . BYTE \$0C, \$0A, \$7B, \$80, \$80, \$0B, \$1E, \$1F CONTROL

IF81 80 80 0B

IF84 1E 1F

IF86 0F 80 10 . BYTE \$0F, \$80, \$10, \$15, \$9B, \$09, \$1C, \$1D

IF89 15 9B 09

IF8C 1C 1D

IF8E 16 80 03 . BYTE \$16, \$80, \$03, \$80, \$80, \$02, \$1B, \$1A

IF91 80 80 02

IF94 1B 1A

IF96 80 80 85 . BYTE \$80, \$80, \$85, \$80, \$1B, \$80, \$FD, \$80

IF99 80 1B 80

IF9C FD 80


```

IF9E 00 20 60      . BYTE  $00, $20, $60, $0E, $80, $0D, $80, $81
IFA1 0E 80 0D
IFA4 80 81
IFA6 12 00 05      . BYTE  $12, $80, $05, $19, $7E, $14, $17, $11
IFA7 19 7E 14
IFAC 17 11

IFAE 80 80 80      . BYTE  $80, $80, $80, $80, $FE, $80, $7D, $FF
IFB1 80 FE 80
IFB4 7D FF
IFB6 06 08 04      . BYTE  $06, $08, $04, $80, $84, $07, $13, $01
IFB9 80 84 07
IFBC 13 01

```

```

IFBE AD 09 D2      PIR0  LDA  KBCODE
IFC1 CD F2 02      CMP  CH1  TEST AGAINST LAST KEY PRESSED
IFC4 D0 05          BNE  PIRQ3 IF NOT, GO PROCESS KEY
IFC6 AD F1 02      LDA  KEYDEL IF KEY DELAY BYTE > 0
IFC9 D0 20          BNE  PIRQ4 IGNORE KEY AS BOUNCE
IFCB AD 09 D2      PIRQ3 LDA  KBCODE RESTORE AC
IFCE C9 9F          CMP  #CNTL1 TEST CONTROL 1 (SSFLA0)
IFD0 D0 0A          BNE  PIRQ1
IFD2 AD FF 02      LDA  SSFLA0
IFD5 49 FF          EOR  #FF
IFD7 8D FF 02      STA  SSFLA0
IFDA 80 0F          BCB  PIRQ4 (UNCONDITIONAL) MAKE 'I' INVISIBLE
FFDC 8D FC 02      PIRQ1 STA  CH
FFDF 8D F2 02      STA  CH1
IFE2 A9 03          LDA  #3
IFE4 8D F1 02      STA  KEYDEL INITIALIZE KEY DELAY FOR DEBOUNCE
IFE7 A9 00          LDA  #0 CLEAR COLOR SHIFT BYTE
FFE9 85 4D          STA  ATRACT
FFEB A9 30          PIRQ4 LDA  #30
FFED 8D 2B 02      STA  BRTMR
IFF0 68            PIRQ2 PLA
IFF1 40            RTI

```

```

IFF2 FF FF FF      . BYTE  $FF, $FF, $FF, $FF, $FF, $FF
IFF3 FF FF FF

```

```

IFFB      CRNTPC  ==
           ==$14

```

```

(0014 00      KBDSPR  . BYTE  $FFFF-CRNTPC  DISPLC IS TOO LONG
           END

```

SYMBOL TABLE

| | | | | | | | |
|---------|------|--------|------|---------|------|--------|------|
| ADDRESS | 030E | ADRESB | 0064 | AFP | D800 | ALLPOT | D208 |
| ALICAT | FE45 | ANCONV | FE69 | ANTIC | D400 | APPEND | 0001 |
| APPMU | 000E | ASCCD1 | F705 | ATACHR | 02FB | ATAINT | FEF4 |
| ATAN | BE43 | ATASCI | FEFE | ATTRACT | 004D | AUDC1 | D201 |
| AUDC2 | D203 | AUDC3 | D205 | AUDC4 | D207 | AUDCTL | D208 |
| AUDF1 | D200 | AUDF2 | D202 | AUDF3 | D204 | AUDF4 | D206 |
| BADMOD | 0086 | BADMOD | 0091 | BELL | F90A | BELL1 | F90C |
| BFENL0 | 0035 | BFENL0 | 0034 | BITCLR | F812 | BITCON | FAEB |
| BITGET | F831 | BITGET | F825 | BITHSK | 006E | BITPUT | F804 |
| BLIM | FAFA | BITSET | F806 | BLIM | 029A | BLKBDV | E471 |
| BOOTAD | 0009 | BOOTAD | 0242 | BOTSCR | 02BF | BPTR | 003D |
| BRKKEY | 0080 | BRKKEY | 0011 | BS | F7E6 | BS1 | F80D |
| BS | F805 | BS | F7E3 | BSS | F7EC | BUFADR | 0015 |
| BUFRFL | 006B | BUFRFL | 0038 | BUFRHI | 0033 | BUFRLO | 0032 |
| CASBUF | 006C | CASBUF | 03FD | CASSETV | E440 | CASFL0 | 030F |
| CASOR0 | 0002 | CASOR0 | EF41 | CASSET | 004B | CASSET | 0043 |
| CAUX1 | 023C | CAUX2 | 023D | CBAUDH | 02EF | CBAUDL | 02EE |
| CDEVID | 023B | CDEVID | 023A | CDTMA1 | 0226 | CDTMA2 | 022B |
| CDTME4 | 022A | CDTME4 | 022C | CDTME5 | 022E | CDTMV1 | 021B |
| CDTMV3 | 021A | CDTMV3 | 021C | CDTMV4 | 021E | CDTMV5 | 0220 |
| CH | 02FC | CH1 | 02F2 | CHACT | 02F3 | CHACTL | D401 |
| CHAS | 02FA | CHBAS | 02F4 | CHBASE | D409 | CHKERR | 008F |
| CHKSUM | 0038 | CHKSUM | 0031 | CHOR0 | E000 | CIOCHR | 002F |
| CIOINV | E46E | CIOOR0 | E4A6 | CIOV | E456 | CIX | 00F2 |
| CLICK | 004A | CLICK | FCDB | CLICK1 | FCDA | CLOBE | 000C |
| CLRL11 | 007D | CLRL11 | FBA6 | CLRLIN | F89B | CLR8C2 | F7BF |
| CLR8CR | F7CE | CLR8CR | F7B9 | CLRTAB | F832 | CLRTB8 | F430 |
| CNTRL8 | 009F | CNTRL8 | FEC6 | COLAC | 0072 | COLBK | D01A |
| COLCR1 | FCE4 | COLCR1 | FCEE | COLCR2 | FCF0 | COLCR6 | 0055 |
| COLDBV | 0244 | COLDBV | E477 | COLINC | 007A | COLOR0 | 02C4 |
| COLDR2 | 02C5 | COLDR2 | 02C6 | COLDR3 | 02C7 | COLDR4 | 02C8 |
| COLPF1 | D016 | COLPF1 | D017 | COLPF2 | D018 | COLPF3 | D019 |
| COLPM0 | D012 | COLPM1 | D013 | COLPM2 | D014 | COLPM3 | D015 |
| COLRTB | 004F | COLRTB | FEC1 | COLUMN | FEBD | COMRE1 | F7A7 |
| CONR1 | F789 | CONR1 | D01F | CONVR1 | F97E | CONVR2 | F988 |
| CONVR3 | F98F | CONVR4 | F99C | CONVR5 | F9A6 | CONVR6 | F9A7 |
| COUNTR | F947 | COS | BD73 | COUNTR | 007E | CR | 009B |
| CRITIC | 0036 | CRITIC | 0042 | CRLOOP | F5E7 | CRNTPC | FFF8 |
| CRSRDN | 02F0 | CRSRDN | F78C | CRSRL1 | F7A3 | CRSRLF | F799 |
| CRSRRT | 008D | CRSRRT | F7AA | CRSRUP | F77F | CROPIV | E47D |
| CTIA | 0288 | CTIA | 0000 | DAUX1 | 030A | DAUX2 | 030B |
| DBDEC | F913 | DBDEC | F91F | DBRECT | 0241 | DBSUB | F921 |
| DBUFH1 | F934 | DBUFH1 | 0305 | DBUFLO | 0304 | DBYTH1 | 0309 |
| DCB | 0308 | DCB | 0300 | DCOMND | 0302 | DDEVIC | 0300 |
| DEGON | 00FB | DEGON | 0006 | DELCH1 | F870 | DELCH2 | F896 |
| DELETE | F86D | DELETE | 0021 | DELL11 | F8DD | DELL12 | F8FB |
| DELL1A | F8D7 | DELL1B | F8D8 | DELL1N | F8D4 | DELTAC | 0077 |
| DELTAR | 0076 | DELT11 | FC82 | DELT12 | FC89 | DELT13 | FC8C |
| DELTIA | FC68 | DELT1B | FC70 | DELT1M | FC73 | DERROR | 0090 |
| DHLINE | 0240 | DHLINE | FEB1 | DIGRT | 00F1 | DINDEX | 0057 |
| DIRECT | 0002 | DISK | 0044 | DISKIV | E450 | DISPLA | E410 |
| DISPLY | 0053 | DIV2TB | FEA5 | DLISTH | D403 | DLISTL | D402 |
| DMASK | D40Q | DMASK | 02A0 | DMASKT | FEB1 | DNACK | 008B |
| DOBU1A | FC1A | DOBU1A | FC29 | DOBUF1 | FC12 | DOBUF2 | FC39 |
| DOBUF3 | FC51 | DOBUF4 | FC55 | DOBUFC | FC00 | DOCR | FA34 |
| DOCR1A | FA00 | DOCR1A | FA29 | DOCR1B | FA14 | DOCR2 | FA3D |
| DOCR2A | FA4A | DOCR2B | FA4D | DOCR4B | FA61 | DOCRWB | FA30 |
| DOLCO1 | FBE5 | DOLCO2 | F8FB | DOLCOL | F8DD | DOPEN | F3F6 |
| DOPEN1 | F460 | DOPEN2 | F4D5 | DOPEN3 | F4FA | DOPEN4 | F51C |

| | | | | | | | |
|--------|------|--------|------|--------|------|--------|------|
| DOPEN5 | F524 | DOPEN7 | F588 | DOPEN8 | F438 | DOPEN9 | F577 |
| DOPENA | F457 | DOSINI | 000C | DQSS | F6AD | DORVEC | 000A |
| DRAW | FCFC | DRAW1 | FD37 | DRAW10 | FE42 | DRAW11 | FD99 |
| DRAW2 | FD5C | DRAW3 | FDB3 | DRAW3A | FD62 | DRAW4A | FD90 |
| DRAW5 | FDA4 | DRAW5A | FDB2 | DRAW6 | FDBE | DRAW6A | FDD7 |
| DRAW8 | FDEB | DRAW7 | FDF1 | DRAW8 | FDF6 | DRAW8A | FE0A |
| DRAW10 | FE27 | DRAW8C | FE13 | DRAW9 | FE30 | DRAWA | FDOC |
| DRAW11 | FDO8 | DRAWLN | Q011 | DRETRY | Q037 | DRMSK | Q04E |
| DSKIN | 001B | DSKINV | E453 | DSKORG | EDEA | DSKTIM | Q246 |
| DSK10 | 001A | DSPFLQ | Q2FE | DSTAT | Q04C | DSTATS | Q303 |
| DLIN | Q306 | DUNIT | Q301 | DUNUSE | Q307 | DVSTAT | Q2FA |
| EDITOR | E400 | EDITRV | E400 | EEXP | Q0ED | EOETC1 | F450 |
| EOETC2 | F46E | EOETC3 | F47C | EOETC5 | F491 | EOETC6 | F46B |
| EOETC7 | F480 | EOETCH | F43E | ENDPT | Q074 | EOERR | Q08B |
| ERREN | F3FC | EOUTC5 | F4BE | EOUTC6 | F4B5 | EOUTCH | F4A4 |
| ERANCE | F48B | ERETN | F4BB | ERRFLQ | Q23F | ESCAPE | F779 |
| ESCLL6 | Q2A2 | ESIGN | Q0FF | EXP | DDC0 | EXP10 | DDCC |
| EXTEN1 | F87E | EXTEN3 | F8BB | EXTEN4 | F892 | EXTEND | F87B |
| FADP | DA66 | FASC | D8E6 | FCHRFL | Q0F0 | FDIV | D82B |
| FLUE | Q03F | FILDAT | Q2FD | FILELQ | Q2B7 | FILLIN | Q012 |
| FLDOR | DD8D | FLDOR | DD89 | FLD1P | DD9C | FLD1R | DD9B |
| FLPTR | Q0FC | FMOVE | DD86 | FMS2P0 | Q043 | FML | DAD8 |
| FNCHIT | Q092 | FORMAT | Q022 | FPI | D9D2 | FPREC | Q006 |
| FPCR | Q5E6 | FPBCR1 | Q5EC | FPTR2 | Q0FE | FRO | Q0D4 |
| FRI | Q0E0 | FR2 | Q0E6 | FRE | Q0DA | FREQ | Q040 |
| FRMADR | Q069 | FRMERR | Q08C | FRX | Q0EC | FRCH | Q5E6 |
| FGRU | Q5EC | FSTOP | DDAB | FSTOR | DDA7 | F8UB | DA60 |
| FTYPE | Q03E | GETCH | F593 | GETCHR | Q007 | GETOUT | F749 |
| GTPLT | F5A2 | GETREC | Q005 | GLBABS | Q2E0 | GPRIOR | Q26F |
| GRACT1 | D01D | GRAFH | D011 | GRAFP0 | D00D | GRAFP1 | D00E |
| GRAFP2 | D00F | GRAFP3 | D010 | HATAB8 | Q31A | HITCLR | D01E |
| HND | Q051 | HOLD2 | Q29F | HOLD3 | Q29D | HOLD4 | Q28C |
| HND2 | Q2BD | HOLDCH | Q07C | HOME | F7D6 | HPOSM0 | D004 |
| HPOBM1 | D005 | HPOBM2 | D006 | HPOBM3 | D007 | HPOBP0 | D000 |
| HPOBP1 | D001 | HPOBP2 | D002 | HPOBP3 | D003 | HSCROL | D404 |
| ICAX1 | Q34A | ICAX12 | Q02A | ICAX2 | Q34B | ICAX22 | Q02B |
| ICBA1 | Q345 | ICBAH2 | Q025 | ICBAL | Q344 | ICBAL2 | Q024 |
| ICBL1 | Q349 | ICBLH2 | Q029 | ICBL | Q34B | ICBL2 | Q02B |
| ICCH | Q342 | ICCOMT | Q017 | ICCOM2 | Q022 | ICDND | Q341 |
| ICDND2 | Q021 | ICHID | Q340 | ICH1D2 | Q020 | ICDND | Q02E |
| ICPT1 | Q347 | ICPTH2 | Q027 | ICPTL | Q346 | ICPTL2 | Q026 |
| ICPR | Q34C | ICSPR2 | Q02C | ICSTA | Q343 | ICSTA2 | Q023 |
| IP | D9AA | INATA1 | F84A | INATAC | F832 | INBUFF | Q0F3 |
| INCPA | F9FB | INCRB1 | FA77 | INCRB2 | F9E4 | INCRB3 | F9F7 |
| INCRBA | F9DC | INCRSB | F9D4 | INCRBC | F9DA | INCRSR | F9DB |
| INSC11 | F852 | INSC13 | F85E | INSC14 | F844 | INSC15 | F86A |
| INSC16 | F861 | INSC1R | F837 | INSC1R | Q020 | INSDAT | Q07D |
| INSL1 | F8C6 | INSL12 | F8CE | INSL1A | F8A5 | INSLIN | F8A4 |
| INTAB0 | Q200 | INTATA | FEFA | INTEMP | Q22D | INTINV | E46B |
| INTLR0 | E6D5 | INTZBB | Q010 | INVFLQ | Q2B6 | IOCB | Q340 |
| IOCBAS | Q020 | IOCB62 | Q010 | IOCFRE | Q0FF | IRGEN | D20E |
| IRGT | D20E | JBRIND | F6A1 | K1 | F729 | K2 | F734 |
| K1 | F73F | K4 | F776 | K5 | F76B | K6 | F74D |
| K7 | F745 | K8 | F773 | KBCDE | D209 | KBD | Q04B |
| KDDIND | E420 | KDDORQ | F3E4 | KBDSR | Q014 | KEYBDV | E420 |
| KEYDEL | Q2F1 | KOETC1 | F71E | KOETC2 | F6DD | KOETC3 | F6FE |
| KCFICH | F6E2 | LBFEND | Q5FF | LBPR1 | Q57E | LBPR2 | Q57F |
| LBUIF | Q580 | LEDGE | Q002 | LFRTCH | F7A5 | LINBUF | Q247 |
| LIN715 | Q000 | LMARGN | Q052 | LO1GET | F822 | LO2GET | F823 |
| LOG11 | Q023 | LOG | DECD | LOG10 | DED1 | LOGICL | Q063 |

| | | | | | | | |
|---------|------|---------|------|--------|------|--------|------|
| LOGOFF | FB20 | LOGMAP | 02B2 | LPENH | 0234 | LPENV | 0235 |
| NOFF | D000 | MOPL | D00B | MIPF | D001 | M1PL | D009 |
| REFF | D002 | M2PL | D00A | M3PF | D003 | M3PL | D00B |
| MAXCIB | FEB9 | MAXDEV | 0021 | MAXIOC | 00B0 | MEMLO | 02E7 |
| MLTMAP | 02E5 | MLTMAP | 0066 | MODEM | 004D | MONDRG | F0E3 |
| MOVLI | FB5B | MOVLI2 | FB7A | MOVLIN | FB4E | MXDMDE | FE5D |
| MXDMID | 0010 | NEWCOL | 0061 | NEWROW | 0060 | NMIEN | D40E |
| NMTIB | D40F | NMIST | D40F | NOCKSM | 003C | NOFUNC | F63D |
| NOROW | F4AB | NONDEV | 00B2 | NOROWB | FE99 | NOBCR1 | FA32 |
| NOSCB | FA2C | NOTB | F4BB | NOTE | 0026 | NOTMXD | F4F5 |
| NOTIBN | 00B5 | NSIGN | 00EE | NUNDLE | FE51 | NVALID | 00B4 |
| OFFERR | FAE4 | OLDADR | 005E | OLDCHR | 005D | OLDCOL | 005B |
| OLDRAW | 005A | OPEN | 0003 | OPNCOM | F404 | OPNERR | F453 |
| OPNIN | 0004 | OPNIND | 000C | OPNOT | 000B | OPNTMP | 0066 |
| OUTCH | F5B7 | OUTCH2 | F5FF | OUTCHA | F5BD | OUTCHB | F5D7 |
| OUTCHF | F5CA | OUTPLT | F5E0 | OVRRUN | 00BE | POPF | D004 |
| POPI | D00C | P1PF | D005 | P1PL | D00D | P2PF | D006 |
| P3PF | D00E | P3PF | D007 | P3PL | D00F | PACTL | D302 |
| PADD0 | 0270 | PADDL1 | 0271 | PADDL2 | 0272 | PADDL3 | 0273 |
| PADD4 | 0274 | PADDL5 | 0275 | PADDL6 | 0276 | PADDL7 | 0277 |
| PACTH | FE75 | PBCTL | D303 | PBPNT | 001D | PBUFSZ | 001E |
| PCOLRO | 02C0 | PCOLR1 | 02C1 | PCOLR2 | 02C2 | PCOLR3 | 02C3 |
| PI-MI | D40C | PENV | D40D | PHACR1 | FC9F | PHACR8 | FC9D |
| PIA | D300 | PIRG | FFBE | PIRG1 | FFDC | PIRG2 | FFF0 |
| PIRG3 | FFCB | PIRG4 | FFEB | PLACR1 | FCAA | PLACR8 | FCAB |
| PLYARG | 05E0 | PLYEVL | DD40 | PMBASE | D407 | POINT | 0025 |
| POKEY | D200 | POKMSK | 0010 | PORTA | D300 | PORTB | D301 |
| POT0 | D200 | POT1 | D201 | POT2 | D202 | POT3 | D203 |
| POT4 | D204 | POT5 | D205 | POT6 | D206 | POT7 | D207 |
| POT00 | D20B | PRINTR | 0050 | PRINTV | E430 | PRIOR | D01B |
| PINBUF | 03C0 | PRNDRO | EE7B | PRVOPN | 00B1 | PTMP | 001F |
| PTIMT | 001C | PTRIG0 | 027C | PTRIG1 | 027D | PTRIG2 | 027E |
| PTRIG3 | 027F | PTRIG4 | 0280 | PTRIG5 | 0281 | PTRIG6 | 0282 |
| PTRIG7 | 0283 | PUTCHR | 000B | PUTMBC | FCF3 | PUTREC | 0009 |
| PWINDA | F3E4 | RADFL0 | 00FB | RADON | 0000 | RAMLO | 0004 |
| RANGE12 | 02E4 | RAMTOP | 006A | RANDON | D20A | RANGE | FA96 |
| RANGE1 | FAB7 | RANGE2 | FAB8 | RANGE3 | FA9E | RBL0KV | E47A |
| RDONLY | 00B7 | RECVDN | 0039 | REDGE | 0027 | RENAME | 0020 |
| RETRRT | F634 | RETR3 | FAE1 | RETURN | F621 | RHARON | 0053 |
| RNGR1 | FAD8 | RNGR2 | FAD6 | RNGERR | FAD1 | RNGOK | FAC4 |
| ROWAC | 0070 | ROWCRS | 0054 | ROWINC | 0079 | RTCLOK | 0012 |
| SAVADR | 006B | SAVID | 0316 | SAVMBC | 005B | SCREDT | 0045 |
| SCRENV | E410 | SCRFLO | 02B8 | SCRHEM | 0093 | SCROL1 | FB87 |
| SCROL2 | FBCA | SCROLL | FBAC | SDLSTH | 0231 | SDLSTL | 0230 |
| SDNCTL | 022F | SENDEV | E46B | SERIN | D20D | SEROUT | D20D |
| SETTAD | FB2D | SETV8V | E45C | SHFANT | 006F | SHFLOK | 02BE |
| SHIF11 | F5B1 | SHIF12 | F610 | SHIF1D | F5AA | SHIF1U | F60B |
| SHIN | BD81 | SHOINV | E465 | SHIDRO | E944 | SHOV | E459 |
| SIZEN | D00C | SIZEPO | D00B | SIZEP1 | D009 | SIZEP2 | D00A |
| SIZEP3 | D00B | SKCTL | D20F | SKRES | D20A | SKSTAT | D20F |
| SOUND | 0041 | SPECIL | 000E | SQR | BEB1 | SRTIMR | 022B |
| STATAC | 02FF | SSKCTL | 0232 | STACKP | 031B | STAT1B | 000D |
| STATUB | 0030 | STICK0 | 027B | STICK1 | 0279 | STICK2 | 027A |
| STICK3 | 027B | STIMER | D209 | STORE | F917 | STORE1 | F91D |
| STRIG0 | FC5C | STRERR | F942 | STRIG0 | 02B4 | STRIG1 | 02B5 |
| STRIG2 | 02B6 | STRIG3 | 02B7 | STROK | F946 | SUBEND | FA7A |
| SUBTMP | 029E | SUCCESS | 0001 | SWAP | FCB3 | SWAP1 | FCC2 |
| SWAP3 | FCD7 | SWAPA | FCB9 | SWPFL0 | 007B | SY8VBV | E45F |
| TAB | FB10 | TAB1 | FB23 | TAB2 | FB2A | TABMAP | 02A3 |
| TEMP | 023E | TEMP1 | 0312 | TEMP2 | 0314 | TEMP3 | 0315 |

DISPATCH HANDLER -- 10-30-78 -- DISPLC

| | | | | | | | |
|---------|------|--------|------|--------|------|--------|------|
| TIMER1 | 030C | TIMER2 | 0310 | TIMFLO | 0317 | TIMOUT | 008A |
| TINDEX | 0293 | IMPCHR | 0050 | IMPCOL | 0289 | IMPLBT | 02A1 |
| THROW | 02B8 | THPX1 | 029C | TOADR | 0066 | TRAMSZ | 0006 |
| TRIG0 | D010 | TRIG1 | D011 | TRIG2 | D012 | TRIG3 | D013 |
| TRNGCP | 0089 | TSTAT | 0319 | TSTCT1 | FC8F | TSTCT2 | FC9C |
| TSTCT1 | FC8D | TSTDAT | 0007 | TXTCOL | 0291 | TXTMSC | 0294 |
| TXTCP | 0296 | TXTROW | 0290 | UNLOCK | 0024 | UPDNCH | F7B7 |
| VBANKA | 0480 | VBREAK | 0206 | VBAIT | F496 | VCRUNT | D40B |
| VCTANI | E480 | VDELAY | D01C | VDSLST | 0200 | VECTBL | E400 |
| VTHK0 | 0216 | VINTER | 0204 | VKEYBD | 0208 | VPRCED | 0202 |
| VZCRCH | D405 | VSERIN | 020A | VSERQC | 020E | VSERDR | 020C |
| VTIMR1 | 0210 | VTIMR2 | 0212 | VTIMR4 | 0214 | VVBLKD | 0224 |
| VVMI KI | 0222 | WARMST | 000B | WARMBV | E474 | WMODE | 02B9 |
| WIDN Y | 00B3 | WSYNC | D40A | XITVBY | E462 | XMTDON | 003A |
| ZIOCH | 0020 | ZTEMP1 | 00F5 | ZTEMP3 | 00F9 | ZTEMP4 | 00F7 |

14127

Report Machine Form, Inc. 87